MPLS Traffic Engineering Path Link and Node Protection Configuration Guide, Cisco IOS XE 3S (Cisco ASR 900 Series)

First Published: 2003-01-16

Last Modified: 2014-07-30

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

CONTENTS

CHAPTER 1

MPLS Traffic Engineering—Fast Reroute Link and Node Protection

The MPLS Traffic Engineering--Fast Reroute Link and Node Protection feature provides link protection (backup tunnels that bypass only a single link of the label-switched path (LSP)), node protection (backup tunnels that bypass next-hop nodes along LSPs), and Fast Reroute (FRR) features.

Prerequisites for MPLS Traffic Engineering—Fast Reroute Link and Node Protection

Your network must support the following Cisco IOS features:

- IP Cisco Express Forwarding
- Multiprotocol Label Switching (MPLS)

Your network must support at least one of the following protocols:

- Intermediate System-to-Intermediate System (IS-IS)
- Open Shortest Path First (OSPF)

Before configuring FRR link and node protection, it is assumed that you have done the following tasks but you do not have to already have configured MPLS traffic engineering (TE) tunnels:

- Enabled MPLS TE on all relevant routers and interfaces
- Configured MPLS TE tunnels

Restrictions for MPLS Traffic Engineering—Fast Reroute Link and Node Protection

- Interfaces must use MPLS Global Label Allocation.
- Backup tunnel headend and tailend routers must implement FRR as described in draft-pan-rsvp-fastreroute-00.txt.
- Backup tunnels are not protected. If an LSP is actively using a backup tunnel and the backup tunnel fails, the LSP is torn down.
- LSPs that are actively using backup tunnels are not considered for promotion. If an LSP is actively using a backup tunnel and a better backup tunnel becomes available, the active LSP is not switched to the better backup tunnel.
- You cannot enable FRR Hellos on a router that also has Resource Reservation Protocol (RSVP) Graceful Restart enabled.
-
- MPLS TE LSPs that are fast reroutable cannot be successfully recovered if the LSPs are FRR active and the Point of Local Repair (PLR) router experiences an SSO.
- When SSO (stateful switchover) occurs on a router, the switchover process must complete before FRR (fast reroute) can complete successfully. In a testing environment, allow approximately 2 minutes for TE SSO recovery to complete before manually triggering FRR. To check the TE SSO status, use the **show ip rsvp high-availability summary** command. Note the status of the HA state field.
  - When SSO is in the process of completing, this field will display 'Recovering'.
  - When the SSO process has completed, this field will display 'Active'.
- It is recommended to configure 10msec of three BFD timers for cable failures, to achieve 50 msec of convergence.

Restrictions for MPLS Traffic Engineering on the Cisco ASR 900 (RSP1 and RSP2) Routers

Starting from Cisco IOS XE Release 3.6S and later, the following restrictions are applicable:

- TE FRR is not supported over Port Channel (PoCH). That is, primary tunnels configured for Link or Node Protection cannot go over port channel interfaces.
- MPLS-TE cutover time of 50 milliseconds is *not* applicable for path protection as it is dependant on the convergence time.
- Starting with Cisco IOS XE Release 3.18, BGP PIC with TDM Pseudowire is supported on the RSP2 module.
- Starting with Cisco IOS XE Relesase 3.18, BGP PIC with MPLS Traffic Engineering for TDM Pseudowires, is supported on the RSP1 and RSP2 modules.

  Restrictions for BGP PIC with MPLS TE for TDM pseudowire:

  - MPLS TE over MLPPP and POS in the core is not supported.
  - Co-existence of BG PPIC with MPLS TE FRR configuration is not supported.

- For tunnel interfaces, tunnel statistics or counter information for each tunnel is *not* supported.
- TE and FRR are *not* supported on POS interfaces on the router.
-

Restrictions for MPLS Traffic Engineering on the RSP3 Module

- Starting with Cisco IOS XE Fuji 16.9.x release, MPLS TE and MPLS-TE FRR is supported over BGP PIC.
- Starting with Cisco IOS XE Fuji 16.9.x release, load balancing of traffic over multiple paths with MPLS-TE FRR is supported.
- Starting with Cisco IOS XE Fuji 16.9.x release, MPLS-TE FRR is supported over labeled BGP (RFC3107).
- MPLS-TE and MPLS-TE FRR is *not* supported over VPLS.
- Starting with Cisco IOS XE Fuji 16.9.x release, MPLS-TE FRR is supported over Port Channel (PoCH).
  - MPLS-TE FRR is supported over PoCH with multiple member links.
  - Configure LACP minimum bundle using the **lacp min-bundle n** command under port channel. The value **n** should be equal to the total number of member links to support TE FRR cutover for 50 msec convergence.
- Starting with Cisco IOS XE Fuji 16.9.x release, for MPLS-TE FRR over Port Channel (PoCH) with multiple member links, when micro-BFD on PoCH is enabled, there is no need for LACP minimum bundle to be equal to the total number of member links. But, without micro-BFD, LACP minimum bundle should be equal to the total number of member links.
- Starting with Cisco IOS XE Fuji 16.9.x release, FRR is supported for:
  - Vanilla IP-FRR
  - BGP-LU IP FRR
  - Max two ECMP Path
  - Non ECMP Path
  - TE-FRR vanilla tunnel
  - TE-FRR labeled tunnel
  - with BFD
  - without BFD
- Multicast over PtoP Tunnel is *not* supported.
- P2MP TE Tunnels is *not* supported.
- MPLS-TE explicit-null is *not* supported.

- Starting with Cisco IOS XE Fuji 16.9.x release, for tunnel interfaces, tunnel statistics or counter information for each tunnel is supported.
- Starting with Cisco IOS XE 3.18.1SP2 and later, MPLS TE Verbatim Path is supported on the RSP3 module.
- Inter AS TE tunnels are *not* supported.
- MPLS TE Autoroute Destination *not* supported for inter-area tunnels.
- MPLS TE Path Protection is *not* supported.
- To avoid high convergence, it is mandatory to configure the 'delay installation' and 'delay cleanup' attributes. Use the following commands to set these parameters:

  mpls traffic-eng reoptimize timers delay installation *30*

  mpls traffic-eng reoptimize timers delay cleanup *60*

  Note The above values are only indicative. Choose any value according to your requirements.

- When LDP is enabled in an MPLS TE setup, for the SSO functionality to work, MPLS LDP NSR must be enabled instead of MPLS LDP GR.
- Starting with Cisco IOS XE Fuji 16.9.x release, targeted LDP (MPLS IP over tunnels) is *not* supported.
- Cisco ASR 900 Router with RSP3 module can push a maximum of 4 MPLS labels in the egress direction. This includes service labels (L3VPN, L2VPN, 6PE/6VPE), RFC 3107 BGP-LU label and RSVP-TE labels for FRR primary or backup paths.

Information About MPLS Traffic Engineering—Fast Reroute Link and Node Protection

This section describes the following:

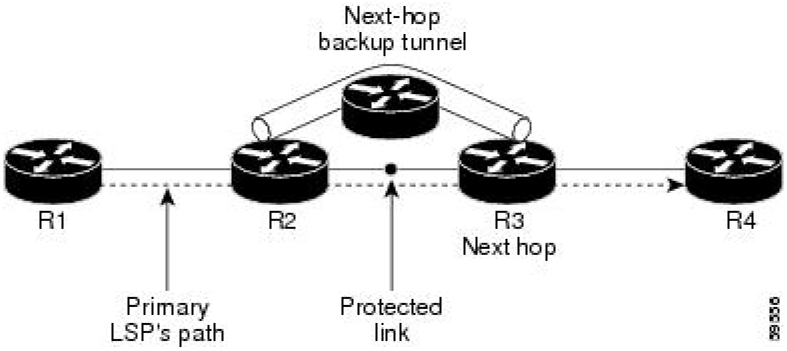
Fast Reroute

Fast Reroute (FRR) is a mechanism for protecting MPLS TE LSPs from link and node failures by locally repairing the LSPs at the point of failure, allowing data to continue to flow on them while their headend routers attempt to establish new end-to-end LSPs to replace them. FRR locally repairs the protected LSPs by rerouting them over backup tunnels that bypass failed links or node.

Link Protection

Backup tunnels that bypass only a single link of the LSP’s path provide link protection. They protect LSPs if a link along their path fails by rerouting the LSP’s traffic to the next hop (bypassing the failed link). These are referred to as next-hop (NHOP) backup tunnels because they terminate at the LSP’s next hop beyond the point of failure. The figure below illustrates an NHOP backup tunnel.

Figure 1: NHOP Backup Tunnel

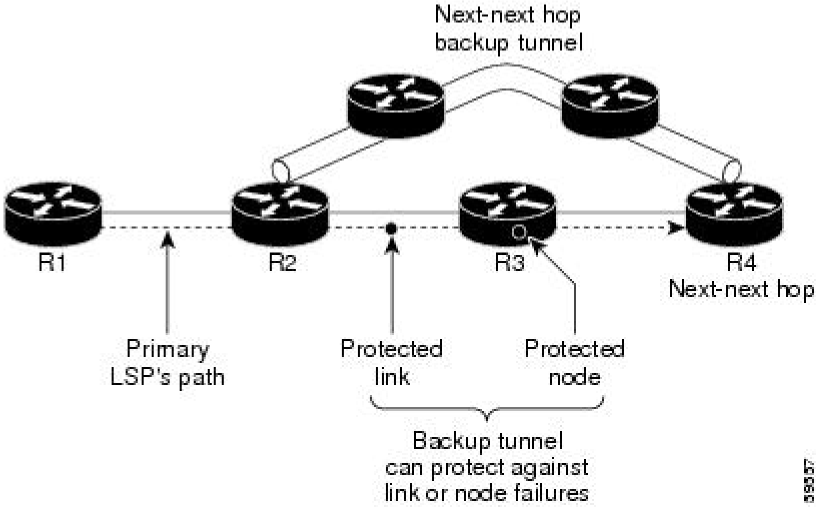


Node Protection

FRR provides node protection for LSPs. Backup tunnels that bypass next-hop nodes along LSP paths are called next-next-hop (NNHOP) backup tunnels because they terminate at the node following the next-hop node of the LSP paths, thereby bypassing the next-hop node. They protect LSPs if a node along their path fails by enabling the node upstream of the failure to reroute the LSPs and their traffic around the failed node to the next-next hop. FRR supports the use of RSVP Hellos to accelerate the detection of node failures. NNHOP backup tunnels also provide protection from link failures, because they bypass the failed link and the node.

The figure below illustrates an NNHOP backup tunnel.

Figure 2: NNHOP Backup Tunnel



If an LSP is using a backup tunnel and something changes so that the LSP is no longer appropriate for the backup tunnel, the LSP is torn down. Such changes are the following:

- Backup bandwidth of the backup tunnel is reduced.
- Backup bandwidth type of backup tunnel is changed to a type that is incompatible with the primary LSP.
- Primary LSP is modified so that FRR is disabled. (The **no mpls traffic-eng fast-reroute** command is entered.)

RSVP Hello

This section describes the following:

RSVP Hello Operation

RSVP Hello enables RSVP nodes to detect when a neighboring node is not reachable. This provides node-to-node failure detection. When such a failure is detected, it is handled in a similar manner as a link-layer communication failure.

RSVP Hello can be used by FRR when notification of link-layer failures is not available (for example, with Ethernet), or when the failure detection mechanisms provided by the link layer are not sufficient for the timely detection of node failures.

A node running Hello sends a Hello Request to a neighboring node every interval. If the receiving node is running Hello, it responds with Hello Ack. If four intervals pass and the sending node has not received an Ack or it receives a bad message, the sending node declares that the neighbor is down and notifies FRR.

There are two configurable parameters:

- Hello interval--Use the **ip rsvp signalling hello refresh interval** command.
- Number of acknowledgment messages that are missed before the sending node declares that the neighbor is down--Use the **ip rsvp signalling hello refresh misses** command

Hello Instance

A Hello instance implements RSVP Hello for a given router interface address and remote IP address. A Hello instance is expensive because of the large number of Hello requests that are sent and the strains they put on the router resources. Therefore, create a Hello instance only when it is necessary and delete it when it is no longer needed.

There are two types of Hello instances:

- Active Hello Instances
- Passive Hello Instances

Active Hello Instances

If a neighbor is unreachable when an LSP is ready to be fast rerouted, an active Hello instance is needed. Create an active Hello instance for each neighbor with at least one LSP in this state.

Active Hello instances periodically send Hello Request messages, and expect Hello Ack messages in response. If the expected Ack message is not received, the active Hello instance declares that the neighbor (remote IP address) is unreachable (lost). LSPs traversing that neighbor may be fast rerouted.

If there is a Hello instance with no LSPs for an unreachable neighbor, do not delete the Hello instance. Convert the active Hello instance to a passive Hello instance because there may be an active instance on the neighboring router that is sending Hello requests to this instance.

Passive Hello Instances

Passive Hello instances respond to Hello Request messages (sending Ack messages), but do not initiate Hello Request messages and do not cause LSPs to be fast rerouted. A router with multiple interfaces can run multiple Hello instances to different neighbors or to the same neighbor.

A passive Hello instance is created when a Hello Request is received from a neighbor with a source IP address/destination IP address pair in the IP header for which a Hello instance does not exist.

Delete passive instances if no Hello messages are received for this instance within 10 minutes.

Features of MPLS Traffic Engineering--Fast Reroute Link and Node Protection

This section describes the following:

Backup Tunnel Support

Backup tunnel support has the following capabilities:

Backup Tunnels Can Terminate at the Next-Next Hop to Support FRR

Backup tunnel that terminates at the next-next hop router to protect both the downstream link and node to protect link and node failures. .

Multiple Backup Tunnels Can Protect the Same Interface

There is no limit (except memory limitations) to the number of backup tunnels that can protect a given interface. In many topologies, support for Node Protection requires supporting multiple backup tunnels per protected interface. These backup tunnels can terminate at the same destination or at different destinations. That is, for a given protected interface, you can configure multiple NHOP or NNHOP backup tunnels. This allows redundancy and load balancing.

In addition to being required for Node Protection, this feature provides the following benefits:

- Redundancy--If one backup tunnel is down, other backup tunnels protect LSPs.
- Increased backup capacity--If the protected interface is a high-capacity link and no single backup path exists with an equal capacity, multiple backup tunnels can protect that one high-capacity link. The LSPs using this link will fail over to different backup tunnels, allowing all of the LSPs to have adequate bandwidth protection during failure (rerouting). If bandwidth protection is not desired, the router spreads LSPs across all available backup tunnels (that is, there is load balancing across backup tunnels). .

Scalability

A backup tunnel is scalable because it can protect multiple LSPs and multiple interfaces. It provides many-to-one (N:1) protection, which has significant scalability advantages over one-to-one (1:1) protection, where a separate backup tunnel must be used for each LSP needing protection.

Example of 1:1 protection: When 5,000 backup tunnels protect 5,000 LSPs, each router along the backup path must maintain state for an additional 5,000 tunnels.

Example of N:1 protection: When one backup tunnel protects 5,000 LSPs, each router along the backup path maintains one additional tunnel.

Fast Reroute Operation

This section describes the following:

Fast Reroute Activation

Two mechanisms cause routers to switch LSPs onto their backup tunnels:

- Interface down notification
- RSVP Hello neighbor down notification

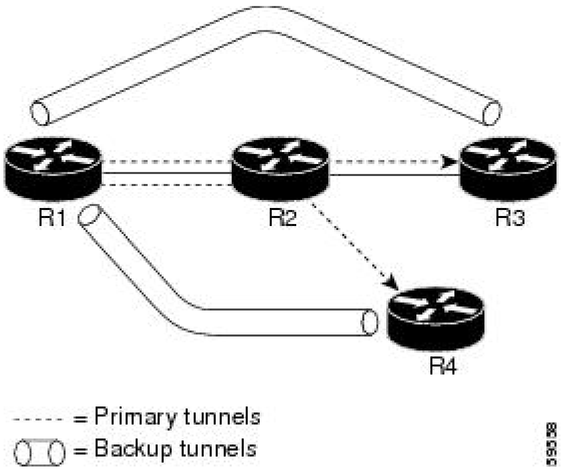
When a router's link or neighboring node fails, the router often detects this failure by an interface down notification. On a GSR Packet over SONET (PoS) interface, this notification is very fast. When a router notices that an interface has gone down, it switches LPSs going out that interface onto their respective backup tunnels (if any).

RSVP Hellos can also be used to trigger FRR. If RSVP Hellos are configured on an interface, messages are periodically sent to the neighboring router. If no response is received, Hellos declare that the neighbor is down. This causes any LSPs going out that interface to be switched to their respective backup tunnels.

Backup Tunnels Terminating at Different Destinations

The figure below illustrates an interface that has multiple backup tunnels terminating at different destinations and demonstrates why, in many topologies, support for node protection requires supporting multiple backup tunnels per protected interface.

Figure 3: Backup Tunnels That Terminate at Different Destinations



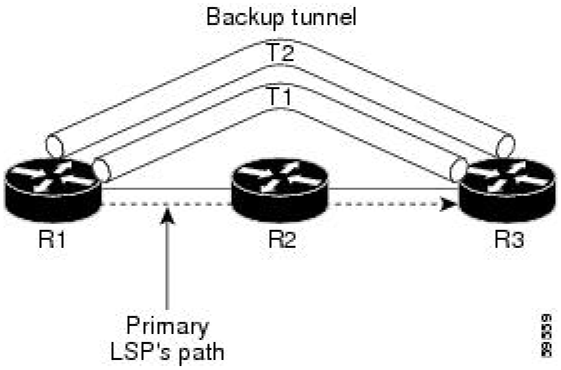
In this illustration, a single interface on R1 requires multiple backup tunnels. LSPs traverse the following routes:

- R1, R2, R3
- R1, R2, R4

To provide protection if node R2 fails, two NNHOP backup tunnels are required: one terminating at R3 and one terminating at R4.

Backup Tunnels Terminating at the Same Destination

The figure below shows how backup tunnels terminating at the same location can be used for redundancy and load balancing. Redundancy and load balancing work for both NHOP and NNHOP backup tunnels.



In this illustration, there are three routers: R1, R2, and R3. At R1 two NNHOP backup tunnels (T1 and T2) go from R1 to R3 without traversing R2.

Redundancy--If R2 fails or the link from R1 to R2 fails, either backup tunnel can be used. If one backup tunnel is down, the other can be used. LSPs are assigned to backup tunnels when the LSPs are first established. This is done before a failure.

Load balancing--If neither backup tunnel has enough bandwidth to back up all LSPs, both tunnels can be used. Some LSPs will use one backup tunnel, other LSPs will use the other backup tunnel. The router decides the best way to fit the LSPs onto the backup tunnels.

Backup Tunnel Selection Procedure

When an LSP is signaled, each node along the LSP path that provides FRR protection for the LSP selects a backup tunnel for the LSP to use if either of the following events occurs:

- The link to the next hop fails.
- The next hop fails.

By having the node select the backup tunnel for an LSP before a failure occurs, the LSP can be rerouted onto the backup tunnel quickly if there is a failure.

For an LSP to be mapped to a backup tunnel, all of the following conditions must exist:

- The LSP is protected by FRR; that is, the LSP is configured with the **tunnel mpls traffic-eng fast-reroute** command.
- The backup tunnel is up.
- The backup tunnel is configured to have an IP address, typically a loopback address.
- The backup tunnel is configured to protect this LSP’s outgoing interface; that is, the interface is configured with the **mpls traffic-eng backup-path** command.
- The backup tunnel does not traverse the LSP’s protected interface.
- The backup tunnel terminates at the LSP’s NHOP or NNHOP. If it is an NNHOP tunnel, it does not traverse the LSP’s NHOP.
- The bandwidth protection requirements and constraints, if any, for the LSP and backup tunnel are met.
  .

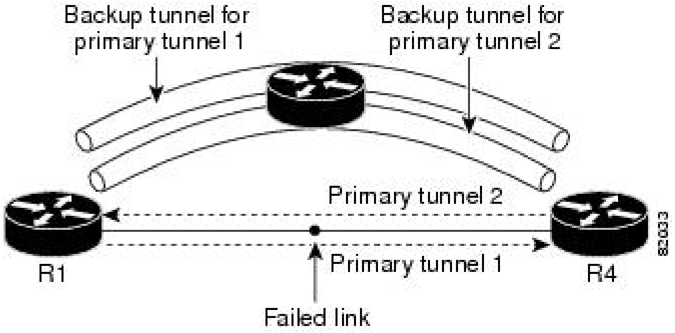
Load Balancing on Limited-Bandwidth Backup Tunnels

There may be more than one backup tunnel that has sufficient backup bandwidth to protect a given LSP. In this case, the router chooses the one that has the least amount of backup bandwidth available. This algorithm limits fragmentation, maintaining the largest amount of backup bandwidth available.

Specifying limited backup bandwidth does not "guarantee" bandwidth protection if there is a link or node failure. For example, the set of NHOP and NNHOP backup tunnels that gets triggered when an interface fails may all share some link on the network topology, and this link may not have sufficient bandwidth to support all LSPs using this set of backup tunnels.

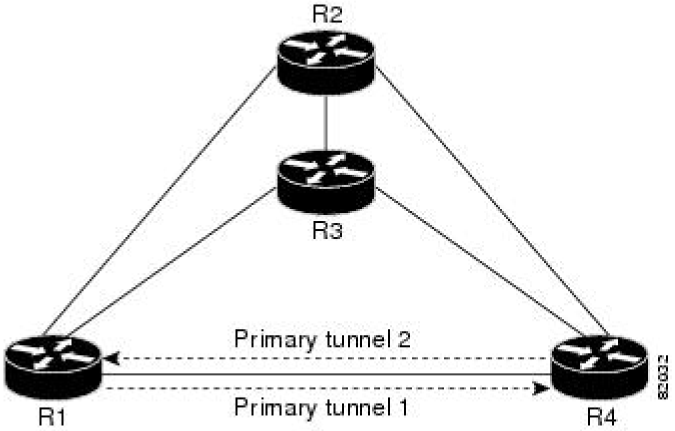
In the figure below, both backup tunnels traverse the same links and hop. When the link between routers R1 and R4 fails, backup tunnels for primary tunnel 1 and primary tunnel 2 are triggered simultaneously. The two backup tunnels may share a link in the network.

Figure 4: Backup Tunnels Share a Link


Backup tunnel for primary tunnel 1
Backup tunnel for primary tunnel 2
Primary tunnel 2
Primary tunnel 1
R1
R4
Failed link
82033

In the figure below, the backup tunnel for primary tunnel 1 may traverse routers R1-R2-R3-R4, and the backup tunnel for primary tunnel 2 may traverse routers R4-R2-R3-R1. In this case, the link R2-R3 may get overloaded if R1-R4 fails.

Figure 5: Overloaded Link



Load Balancing on Unlimited-Bandwidth Backup Tunnels

More than one backup tunnel, each having unlimited backup bandwidth, can protect a given interface. In this case, when choosing a backup tunnel for a given LSP, the router chooses the backup tunnel that has the least amount of backup bandwidth in use. This algorithm evenly distributes the LSPs across backup tunnels based on an LSP's bandwidth. If an LSP is requesting zero bandwidth, the router chooses the backup tunnel that is protecting the fewest LSPs.

Tunnel Selection Priorities

This section describes the following:

NHOP Versus NNHOP Backup Tunnels

More than one backup tunnel can protect a given LSP, where one backup tunnel terminates at the LSP's NNHOP, and the other terminates at the LSP's NHOP. In this case, the router chooses the backup tunnel that terminates at the NNHOP (that is, FRR prefers NNHOP over NHOP backup tunnels).

The table below lists the tunnel selection priorities. The first choice is an NNHOP backup tunnel that acquires its bandwidth from a subpool or global pool, and has limited bandwidth. If there is no such backup tunnel, the next choice (2) is a next-next hop backup tunnel that acquires a limited amount of bandwidth from any pool. The preferences go from 1 (best) to 8 (worst), where choice 3 is for an NNHOP backup tunnel with an unlimited amount of subpool or global-pool bandwidth.

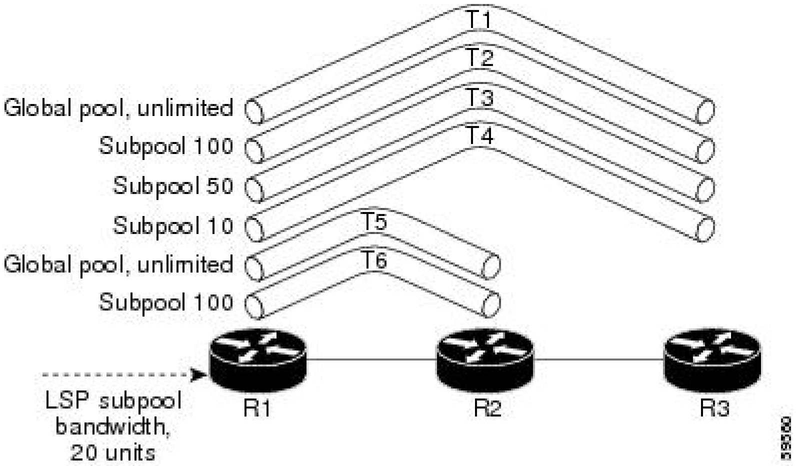
Table 1: Tunnel Selection Priorities

Preference	Backup Tunnel Destination	Bandwidth Pool	Bandwidth Amount
1 (Best)	NNHOP	Subpool or global pool	Limited
2	NNHOP	Any	Limited
3	NNHOP	Subpool or global pool	Unlimited
4	NNHOP	Any	Unlimited
5	NHOP	Subpool or global pool	Limited
6	NHOP	Any	Limited
7	NHOP	Subpool or global pool	Unlimited
8 (Worst)	NHOP	Any	Unlimited

The figure below shows an example of the backup tunnel selection procedure based on the designated amount of global pool and subpool bandwidth currently available.

Note If NHOP and NNHOP backup tunnels do not have sufficient backup bandwidth, no consideration is given to the type of data that the LSP is carrying. For example, a voice LSP may not be protected unless it is signaled before a data LSP. To prioritize backup tunnel usage, see the "Backup Protection Preemption Algorithms" section.

Figure 6: Choosing from Among Multiple Backup Tunnels



In this example, an LSP requires 20 units (kilobits per second) of sub-pool backup bandwidth. The best backup tunnel is selected as follows:

1. Backup tunnels T1 through T4 are considered first because they terminate at the NNHOP.
2. Tunnel T4 is eliminated because it has only ten units of sub-pool backup bandwidth.
3. Tunnel T1 is eliminated because it protects only LSPs using global-pool bandwidth.
4. Tunnel T3 is chosen over T2 because, although both have sufficient backup bandwidth, T3 has the least backup bandwidth available (leaving the most backup bandwidth available on T2).
5. Tunnels T5 and T6 need not be considered because they terminate at an NHOP, and therefore are less desirable than T3, which terminates at an NNHOP.

Promotion

After a backup tunnel has been chosen for an LSP, conditions may change that will cause us to reevaluate this choice. This reevaluation, if successful, is called promotion. Such conditions may include:

1. A new backup tunnel comes up.
2. The currently chosen backup tunnel for this LSP goes down.
3. A backup tunnel’s available backup bandwidth increases. For example, an LSP protected by the tunnel has been reoptimized by the headend to use another path.

For cases 1 and 2, the LSP’s backup tunnel is evaluated immediately. Case 3 is addressed by periodically reevaluating LSP-to-backup tunnel mappings. By default, background reevaluation is performed every 5 minutes. This interval is configurable via the **mpls traffic-eng fast-reroute timers** command.

Backup Protection Preemption Algorithms

When you set the “bandwidth protection desired” bit for an LSP, the LSP has a higher right to select backup tunnels that provide bandwidth protection and it can preempt other LSPs that do not have that bit set.

If there is insufficient backup bandwidth on NNHOP backup tunnels but not on NHOP backup tunnels, the bandwidth-protected LSP does not preempt NNHOP LSPs; it uses NHOP protection.

If there are multiple LSPs using a given backup tunnel and one or more must be demoted to provide bandwidth, there are two user-configurable methods (algorithms) that the router can use to determine which LSPs are demoted:

- Minimize amount of bandwidth that is wasted.
- Minimize the number of LSPs that are demoted.

For example, If you need ten units of backup bandwidth on a backup tunnel, you can demote one of the following:

- A single LSP using 100 units of bandwidth--Makes available more bandwidth than needed, but results in lots of waste
- Ten LSPs, each using one unit of bandwidth--Results in no wasted bandwidth, but affects more LSPs

The default algorithm is to minimize the number of LSPs that are demoted. To change the algorithm to minimize the amount of bandwidth that is wasted, enter the **mpls traffic-eng fast-reroute backup-prot-preemption optimize-bw** command.

Bandwidth Protection Considerations

There are numerous ways in which bandwidth protection can be ensured. The table below describes the advantages and disadvantages of three methods.

Table 2: Bandwidth Protection Methods

Method	Advantages	Disadvantages
Reserve bandwidth for backup tunnels explicitly.	It is simple.	It is a challenge to allow bandwidth sharing of backup tunnels protecting against independent failures.
Use backup tunnels that are signaled with zero bandwidth.	It provides a way to share bandwidth used for protection against independent failures, so it ensures more economical bandwidth usage.	It may be complicated to determine the proper placement of zero bandwidth tunnels.
Backup bandwidth protection.	It ensures bandwidth protection for voice traffic.	An LSP that does not have backup bandwidth protection can be demoted at any time if there is not enough backup bandwidth and an LSP that has backup bandwidth protection needs bandwidth.

Cisco implementation of FRR does not mandate a particular approach, and it provides the flexibility to use any of the above approaches. However, given a range of configuration choices, be sure that the choices are constant with a particular bandwidth protection strategy.

The following sections describe some important issues in choosing an appropriate configuration:

Using Backup Tunnels with Explicitly Signaled Bandwidth

Two bandwidth parameters must be set for a backup tunnel:

- Actual signaled bandwidth
- Backup bandwidth

To signal bandwidth requirements of a backup tunnel, configure the bandwidth of the backup tunnel by using the **tunnel mpls traffic-eng bandwidth** command.

To configure the backup bandwidth of the backup tunnel, use the **tunnel mpls traffic-eng backup-bw** command.

The signaled bandwidth is used by the LSRs on the path of the backup tunnel to perform admission control and do appropriate bandwidth accounting.

The backup bandwidth is used by the point of local repair (PLR) (that is, the headend of the backup tunnel) to decide how much primary traffic can be rerouted to this backup tunnel if there is a failure.

Both parameters need to be set to ensure proper operation. The numerical value of the signaled bandwidth and the backup bandwidth should be the same.

Protected Bandwidth Pools and the Bandwidth Pool from Which the Backup Tunnel Reserves Its Bandwidth

The **tunnel mpls traffic-eng bandwidth** command allows you to configure the following:

- Amount of bandwidth a backup tunnel reserves
- The DS-TE bandwidth pool from which the bandwidth needs to be reserved

Note Only one pool can be selected (that is, the backup tunnel can explicitly reserve bandwidth from either the global pool or the subpool, but not both).

The **tunnel mpls traffic-eng backup-bw** command allows you to specify the bandwidth pool to which the traffic must belong for the traffic to use this backup tunnel. Multiple pools are allowed.

There is no direct correspondence between the bandwidth pool that is protected and the bandwidth pool from which the bandwidth of the backup tunnel draws its bandwidth.

Bandwidth protection for 10 Kbps of subpool traffic on a given link can be achieved by configuring any of the following command combinations:

- **tunnel mpls traffic-eng bandwidth sub-pool 10**

tunnel mpls traffic-eng backup-bw sub-pool 10

- **tunnel mpls traffic-eng bandwidth global-pool 10**

tunnel mpls traffic-eng backup-bw sub-pool 10 global-pool unlimited

- **tunnel mpls traffic-eng bandwidth global-pool 40**

tunnel mpls traffic-eng backup-bw sub-pool 10 global-pool 30

Using Backup Tunnels Signaled with Zero Bandwidth

Frequently it is desirable to use backup tunnels with zero signaled bandwidth, even when bandwidth protection is required. It may seem that if no bandwidth is explicitly reserved, no bandwidth guarantees can be provided. However, that is not necessarily true.

In the following situation:

- Only link protection is desired.
- Bandwidth protection is desired only for sub-pool traffic.

For each protected link AB with a maximum reservable subpool value of *n* , there may be a path from node A to node B such that the difference between the maximum reservable global and the maximum reservable subpool is at least the value of *n* . If it is possible to find such paths for each link in the network, you can establish all the backup tunnels along such paths without any bandwidth reservations. If there is a single link failure, only one backup tunnel will use any link on its path. Because that path has at least *n* available bandwidth (in the global pool), assuming that marking and scheduling is configured to classify the subpool traffic into a priority queue, the subpool bandwidth is guaranteed.

This approach allows sharing of the global pool bandwidth between backup tunnels protecting independent link failures. The backup tunnels are expected to be used for only a short period of time after a failure (until the headends of affected LSPs reroute those LSPs to other paths with available subpool bandwidth). The probability of multiple unrelated link failures is very small (in the absence of node or shared risk link group (SRLG) failures, which result in multiple link failures). Therefore, it is reasonable to assume that link failures are in practice independent with high probability. This “independent failure assumption” in combination with backup tunnels signaled without explicit bandwidth reservation enables efficient bandwidth sharing that yields substantial bandwidth savings.

Backup tunnels protecting the subpool traffic do now draw bandwidth from any pool. Primary traffic using the global pool can use the entire global pool, and primary traffic using the subpool can use the entire subpool. Yet, subpool traffic has a complete bandwidth guarantee if there is a single link failure.

A similar approach can be used for node and SRLG protection. However, the decision of where to put the backup tunnels is more complicated because both node and SRLG failures effectively result in the simultaneous failure of several links. Therefore, the backup tunnels protecting traffic traversing all affected links cannot be computed independently of each other. The backup tunnels protecting groups of links corresponding to different failures can still be computed independently of each other, which results in similar bandwidth savings.

Signaled Bandwidth Versus Backup Bandwidth

Backup bandwidth is used locally (by the router that is the headend of the backup tunnel) to determine which, and how many, primary LSPs can be rerouted on a particular backup tunnel. The router ensures that the combined bandwidth requirement of these LSPs does not exceed the backup bandwidth.

Therefore, even when the backup tunnel is signaled with zero bandwidth, the backup bandwidth must be configured with the value corresponding to the actual bandwidth requirement of the traffic protected by this backup tunnel. Unlike the case when bandwidth requirements of the backup tunnels are explicitly signaled, the value of the signaled bandwidth (which is zero) is not the same value as the backup bandwidth.

How to Configure MPLS Traffic Engineering—Fast Reroute Link and Node Protection

This section assumes that you want to add FRR protection to a network in which MPLS TE LSPs are configured.

Enabling Fast Reroute on LSPs

LSPs can use backup tunnels only if they have been configured as fast reroutable. To do this, enter the following commands at the headend of each LSP.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *number*
4. **tunnel mpls traffic-eng fast-reroute** [**bw-protect**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface tunnel** *number* **Example:** `Router(config)# interface tunnel 1000`	Enters interface configuration mode for the specified tunnel.
Step 4	**tunnel mpls traffic-eng fast-reroute** [**bw-protect**] **Example:** `Router(config-if)# tunnel mpls traffic-eng fast-reroute bw-protect`	Enables an MPLS TE tunnel to use an established backup tunnel if there is a link or node failure.

Creating a Backup Tunnel to the Next Hop or to the Next-Next Hop

Creating a backup tunnel is basically no different from creating any other tunnel. To create a backup tunnel to the next hop or to the next-next hop, enter the following commands on the node that will be the headend of the backup tunnel (that is, the node whose downstream link or node may fail). The node on which you enter these commands must be a supported platform. See the Finding Feature Information section.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *number*
4. **ip unnumbered** *interface-type interface-number*
5. **tunnel destination** *ip-address*
6. **tunnel mode mpls traffic-eng**
7. **tunnel mpls traffic-eng path-option** [**protect**] *preference-number*{**dynamic** | **explicit**|{**name** *path-name* | *path-number*}**verbatim**}[**lockdown**]
8. **ip explicit-path name** *word*
9. **exclude-address** *ip-address*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface tunnel** *number* **Example:** `Router(config)# interface tunnel 1`	Creates a new tunnel interface and enters interface configuration mode.
Step 4	**ip unnumbered** *interface-type interface-number* **Example:** `Router(config-if)# ip unnumbered loopback 0`	Gives the tunnel interface an IP address that is the same as that of interface Loopback0. **Note** This command is not effective until Lookback0 has been configured with an IP address.
Step 5	**tunnel destination** *ip-address* **Example:** `Router(config-if)# tunnel destination 10.3.3.3`	Specifies the IP address of the device where the tunnel will terminate. This address should be the router ID of the device that is the NHOP or NNHOP of LSPs to be protected.
Step 6	**tunnel mode mpls traffic-eng** **Example:** `Router(config-if)# tunnel mode mpls traffic-eng`	Sets the encapsulation mode of the tunnel to MPLS TE.

	Command or Action	Purpose
Step 7	**tunnel mpls traffic-eng path-option** [**protect**] *preference-number*{**dynamic** | **explicit**|{**name** *path-name* | *path-number*}**verbatim**}[**lockdown**] **Example:** `Router(config-if)# tunnel mpls traffic-eng path-option 10 explicit avoid-protected-link`	Configures a path option for an MPLS TE tunnel. Enters router configuration mode.
Step 8	**ip explicit-path name** *word* **Example:** `Router(config-router)# ip explicit-path name avoid-protected-link`	Enters the command mode for IP explicit paths and creates the specified path. Enters explicit path command mode.
Step 9	**exclude-address** *ip-address* **Example:** `Router(config-ip-expl-path)# exclude-address 3.3.3.3`	For link protection, specify the IP address of the link to be protected. For node protection, specify the router ID of the node to be protected. **Note** Backup tunnel paths can be dynamic or explicit and they do not have to use exclude-address. Because backup tunnels must avoid the protected link or node, it is convenient to use the **exclude-address** command. **Note** When using the **exclude-address** command to specify the path for a backup tunnel, you must exclude an interface IP address to avoid a link (for creating an NHOP backup tunnel), or a router ID address to avoid a node (for creating an NNHOP backup tunnel).

Assigning Backup Tunnels to a Protected Interface

To assign one or more backup tunnels to a protected interface, enter the following commands on the node that will be the headend of the backup tunnel (that is, the node whose downstream link or node may fail). The node on which you enter these commands must be a supported platform. See the Finding Feature Information section.

Note You must configure the interface to have an IP address and to enable the MPLS TE tunnel feature.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type slot* / *port*
4. **mpls traffic-eng backup-path tunnel** *interface*

DETAILED STEPS

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

Step 3	**interface** *type slot* / *port*
	Example:
	`Router(config)# interface POS 5/0`
	For RSP3:
	`Router(config)# interface TenGigabitEthernet0/3/0`
		Moves configuration to the physical interface level, directing subsequent configuration commands to the specific physical interface identified by the *type*value. The *slot* and *port* identify the slot and port being configured. The interface must be a supported interface. See the Finding Feature Information section. Enters interface configuration mode.

Step 4	**mpls traffic-eng backup-path tunnel** *interface* **Example:** `Router(config-if)# mpls traffic-eng backup-path tunnel 2`	Allows LSPs going out this interface to use this backup tunnel if there is a link or node failure. **Note** You can enter this command multiple times to associate multiple backup tunnels with the same protected interface.

Associating Backup Bandwidth and Pool Type with a Backup Tunnel

To associate backup bandwidth with a backup tunnel and designate the type of LSP that can use a backup tunnel, enter the following commands.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *number*
4. **tunnel mpls traffic-eng backup-bw** {*bandwidth* | [**sub-pool** {*bandwidth* | **Unlimited**}] [**global-pool** {*bandwidth* | **Unlimited**}]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	`Router> enable`	
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface tunnel** *number* **Example:** `Router(config)# interface tunnel 2`	Enters interface configuration mode for the specified tunnel.
Step 4	**tunnel mpls traffic-eng backup-bw** {*bandwidth* | [**sub-pool** {*bandwidth* | **Unlimited**}] [**global-pool** {*bandwidth* | **Unlimited**}] **Example:** `Router(config-if)# tunnel mpls traffic-eng backup-bw sub-pool 1000`	Associates bandwidth with a backup tunnel and designates whether LSPs that allocate bandwidth from the specified pool can use the tunnel.

Configuring Backup Bandwidth Protection

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *number*
4. **tunnel mpls traffic-eng fast-reroute** [**bw-protect**]
5. **mpls traffic-eng fast-reroute backup-prot-preemption** [**optimize-bw**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters interface configuration mode.
Step 3	**interface tunnel** *number* **Example:**	Configures a new tunnel interface.

	Command or Action	Purpose
	`Router(config)# interface tunnel 1`	
Step 4	**tunnel mpls traffic-eng fast-reroute** [**bw-protect**] **Example:** `Router(config-if)# tunnel mpls traffic-eng fast-reroute bw-protect`	Enables an MPLS TE tunnel to use an established backup tunnel in the event of a link or node failure. • The **bw-protect** keyword gives an LSP priority for using backup tunnels with bandwidth protection. Enters global configuration mode.
Step 5	**mpls traffic-eng fast-reroute backup-prot-preemption** [**optimize-bw**] **Example:** `Router(config-if)# mpls traffic-eng fast-reroute backup-prot-preemption optimize-bw`	Changes the backup protection preemption algorithm from minimize the number of LSPs that are demoted to minimize the amount of bandwidth that is wasted.

Verifying That Fast Reroute Is Operational

To verify that FRR can function, perform the following task.

SUMMARY STEPS

1. **show mpls traffic-eng tunnels brief**
2. **show ip rsvp sender detail**
3. **show mpls traffic-eng fast-reroute database**
4. **show mpls traffic-eng tunnels backup**
5. **show mpls traffic-eng fast-reroute database**
6. **show ip rsvp reservation**

DETAILED STEPS

Step 1 **show mpls traffic-eng tunnels brief**

Use this command to verify that backup tunnels are up:

Example:

```
Router# show mpls traffic-eng tunnels brief

Signalling Summary:
    LSP Tunnels Process:           running
    RSVP Process:                  running
    Forwarding:                    enabled
    Periodic reoptimization:       every 3600 seconds, next in 1706 seconds
TUNNEL NAME                      DESTINATION      UP IF     DOWN IF   STATE/PROT
Router_t1                        10.112.0.12      -         PO2/0/1   up/up
Router_t2                        10.112.0.12      -         unknown   up/down
Router_t3                        10.112.0.12      -         unknown   admin-down
Router_t1000                     10.110.0.10      -         unknown   up/down
Router_t2000                     10.110.0.10      -         PO2/0/1   up/up
Displayed 5 (of 5) heads, 0 (of 0) midpoints, 0 (of 0) tails
```

Step 2 **show ip rsvp sender detail**

Use this command to verify that LSPs are protected by the appropriate backup tunnels.

Following is sample output from the **show ip rsvp sender detail** command when the command is entered at the PLR before a failure:

Example:

```
Router# show ip rsvp sender detail

PATH:
 Tun Dest:   10.10.0.6  Tun ID: 100  Ext Tun ID: 10.10.0.1
 Tun Sender: 10.10.0.1  LSP ID: 31
 Path refreshes:
  arriving: from PHOP 10.10.7.1 on FE0/0/0 every 30000 msecs
 Session Attr:
  Setup Prio: 7, Holding Prio: 7
  Flags: (0x7) Local Prot desired, Label Recording, SE Style
  session Name: R1_t100
 ERO: (incoming)
  10.10.7.2 (Strict IPv4 Prefix, 8 bytes, /32)
  10.10.0.6 (Strict IPv4 Prefix, 8 bytes, /32)
 RRO:
   10.10.7.1/32, Flags:0x0 (No Local Protection)
   10.10.4.1/32, Flags:0x9 (Local Prot Avail/to NNHOP) !Available to NNHOP
   10.10.1.1/32, Flags:0x0 (No Local Protection)
 Traffic params - Rate: 10K bits/sec, Max. burst: 1K bytes
   Min Policed Unit: 0 bytes, Max Pkt Size 4294967295 bytes
 Fast-Reroute Backup info:
   Inbound  FRR: Not active
   Outbound FRR: No backup tunnel selected
 Path ID handle: 50000416.
 Incoming policy: Accepted. Policy source(s): MPLS/TE
 Status: Proxy-terminated
```

Step 3 **show mpls traffic-eng fast-reroute database**

Enter the **clear ip rsvp hello instance counters** command to verify the following:

- MPLS TE FRR Node Protection has been enabled.
- A certain type of LSP can use a backup tunnel.

The following command output displays the LSPs that are protected:

Example:

```
Router# show mpls traffic-eng fast-reroute database
Tunnel head end item frr information:
Protected tunnel              In-label Out intf/label     FRR intf/label   Status
Tunnel500                     Tun hd   AT2/0/0.100:Untagg Tu501:20         ready
Prefix item frr information:
Prefix             Tunnel     In-label Out intf/label     FRR intf/label   Status
10.0.0.8/32        Tu500      18       AT2/0/0.100:Pop ta Tu501:20          ready
10.0.8.8/32        Tu500      19       AT2/0/0.100:Untagg Tu501:20          ready
10.8.9.0/24        Tu500      22       AT2/0/0.100:Untagg Tu501:20          ready
LSP midpoint item frr information:
LSP identifier                In-label Out intf/label   FRR intf/label   Status
```

If LDP is not enabled, separate prefix items are not shown because all prefixes then use a single rewrite. To confirm that a particular IP prefix is FRR protected, even though it is not shown in this display, enter it within the **show mpls forwarding-table** *ip-address* **detail** command. The final line of the display will tell whether that prefix is protected:

Example:

```
Router# show mpls forwarding-table 10.0.0.11 32 detail

Local    Outgoing    Prefix         Bytes tag    Outgoing     Next Hop
tag      tag or VC   or Tunnel Id   switched     interface
Tun hd   Untagged    10.0.0.11/32   48           gigabitethernet1/0/0
      point2point
          MAC/Encaps=4/8, MTU=1520, Tag Stack{22}
          48D18847 00016000
          No output feature configured
          Fast Reroute Protection via (Tu0, outgoing label 12304)
```

The following command output displays the LSPs that are protected when the FRR *backup* tunnel is over an ATM interface:

Example:

```
Router# show mpls traffic-eng fast-reroute database

Tunnel head end item frr information:
Protected tunnel In-label   Out intf/label FRR intf/label Status
Tunnel500 Tun hd PO0/2/0:Untagged Tu501:20 ready
Prefix item frr information:
Prefix Tunnel In-label Out intf/label FRR intf/label Status
10.0.0.8/32 Tu500 18 PO0/2/0:Pop tag  Tu501:20 ready
10.0.8.8/32 Tu500 19 PO0/2/0:Untagged Tu501:20 ready
10.8.9.0/24 Tu500 22 PO0/2/0:Untagged Tu501:20 ready
LSP midpoint item frr information:
LSP identifier In-label Out intf/label FRR intf/label Status
```

Step 4 **show mpls traffic-eng tunnels backup**

The following conditions must exist for backup tunnels to be operational:

- **LSP is reroutable** --At the headend of the LSP, enter the **show run int tunnel** *tunnel-number* command. The output should include the **tunnel mpls traffic-eng fast-reroute** command. If it does not, enter this command for the tunnel.

On the router where the backup tunnels originate, enter the **show mpls traffic-eng tunnels backup**command. Following is sample command output:

Example:

```
Router# show mpls traffic-eng tunnels backup
Router_t578
  LSP Head, Tunnel578, Admin: up, Oper: up
  Src 10.55.55.55, Dest 10.88.88.88, Instance 1
  Fast Reroute Backup Provided:
    Protected i/fs:   gig1/0/0
    Protected lsps: 1
    Backup BW: any pool unlimited; inuse: 100 kbps
Router_t5710
  LSP Head, Tunnel5710, Admin: admin-down, Oper: down
  Src 10.55.55.55, Dest 10.7.7.7, Instance 0
  Fast Reroute Backup Provided:
    Protected i/fs:  gig1/0/0
    Protected lsps: 0
    Backup BW: any pool unlimited; inuse: 0 kbps
Router_t5711
  LSP Head, Tunnel5711, Admin: up, Oper: up
  Src 10.55.55.55, Dest 10.7.7.7, Instance 1
  Fast Reroute Backup Provided:
    Protected i/fs:  gig1/0/0
    Protected lsps: 2
    Backup BW: any pool unlimited; inuse: 6010 kbps
```

The command output will allow you to verify the following:

- Backup tunnel exists--Verify that there is a backup tunnel that terminates at this LSP's NHOP or NNHOP. Look for the LSP's NHOP or NNHOP in the Dest field.
- Backup tunnel is up--To verify that the backup tunnel is up, look for "Up" in the State field.
- Backup tunnel is associated with LSP's I/F--Verify that the interface for the LSP is allowed to use this backup tunnel. Look for the LSP's output interface in the "protects" field list.
- Backup tunnel has sufficient bandwidth--If you restricted the amount of bandwidth a backup tunnel can hold, verify that the backup tunnel has sufficient bandwidth to hold the LSPs that would use this backup tunnel if there is a failure. The bandwidth of an LSP is defined by the line **tunnel mpls traffic-eng bandwidth** at the headend of the LSP. To determine the available bandwidth on a backup tunnel, look at the "cfg" and "inuse" fields. If there is insufficient backup bandwidth to accommodate the LSPs that would use this backup tunnel in the event of a failure, create an additional backup tunnel or increase the backup bandwidth of the existing tunnel by using the **tunnel mpls traffic-eng bandwidth** command.

Note To determine how much bandwidth is sufficient, offline capacity planning may be required.

- Backup tunnel has appropriate bandwidth type--If you restricted the type of LSPs (subpool or global pool) that can use this backup tunnel, verify that the LSP is the appropriate type for the backup tunnel. The type of the LSP is defined by the line **tunnel mpls traffic-eng bandwidth** at the headend of this LSP. If this line contains the word "subpool", then it uses subpool bandwidth; otherwise, it uses global pool bandwidth. Verify that the type matches the type the backup tunnel can hold by looking in the output of the above command.

If none of the above actions works, enable debug by entering the **debug ip rsvp fast-reroute** command and the **debug mpls traffic-eng fast-reroute** command on the router that is the headend of the backup tunnel. Then do the following:

a. Enter the **shutdown** command for the primary tunnel.

b. Enter the **no shutdown** command for the primary tunnel.

c. View the debug output.

Step 5 **show mpls traffic-eng fast-reroute database**

Enter the **clear ip rsvp hello instance counters** command to verify the following:

- MPLS TE FRR Node Protection has been enabled.
- A certain type of LSP can use a backup tunnel.

The following command output displays the LSPs that are protected:

Example:

```
Router# show mpls traffic-eng fast-reroute database
Tunnel head end item frr information:
```

```
Protected Tunnel   In-label   intf/label         FRR intf/label      Status
Tunnel10           Tun         gig1/0/0:Untagged  Tu0:12304           ready
Prefix item frr information:
Prefix        Tunnel  In-label    Out intf/label     FRR intf/label Status
10.0.0.11/32  Tu110   Tun hd       gig1/0/0:Untagged  Tu0:12304      ready
LSP midpoint frr information:
LSP identifier       In-label Out intf/label     FRR intf/label   Status
10.0.0.12 1 [459]    16        gig1/0/0:17        Tu2000:19        ready
```

Note If LDP is not enabled, separate prefix items are not shown because all prefixes then use a single rewrite. To confirm that a particular IP prefix is FRR protected, even though it is not shown in this display, enter it within the **show mpls forwarding-table** *ip-address* **detail** command. The final line of the display will tell whether that prefix is protected:

Example:

```
Router# show mpls forwarding-table 10.0.0.11 32 detail

Local    Outgoing    Prefix         Bytes tag    Outgoing       Next Hop
tag      tag or VC   or Tunnel Id   switched     interface
Tun hd   Untagged    10.0.0.11/32   48            gig1/0/0       point2point
          MAC/Encaps=4/8, MTU=1520, Tag Stack{22}
          48D18847 00016000
          No output feature configured
          Fast Reroute Protection via (Tu0, outgoing label 12304)
```

Step 6 **show ip rsvp reservation**

Following is sample output from the **show ip rsvp reservation** command entered at the headend of a primary LSP. Entering the command at the head-end of the primary LSP shows, among other things, the status of FRR (that is, local protection) at each hop this LSP traverses. The per-hop information is collected in the Record Route Object (RRO) that travels with the Resv message from the tail to the head.

Example:

```
Router# show ip rsvp reservation detail
Reservation:
  Tun Dest:   10.1.1.1  Tun ID: 1  Ext Tun ID: 10.1.1.1
  Tun Sender: 10.1.1.1  LSP ID: 104
  Next Hop: 10.1.1.2 on  gig1/0/0
  Label: 18 (outgoing)
  Reservation Style is Shared-Explicit, QoS Service is Controlled-Load
  Average Bitrate is 0 bits/sec, Maximum Burst is 1K bytes
  Min Policed Unit: 0 bytes, Max Pkt Size: 0 bytes
  RRO:
    10.1.1.1/32, Flags:0x1 (Local Prot Avail/to NHOP)
      Label subobject: Flags 0x1, C-Type 1, Label 18
    10.1.1.1/32, Flags:0x0 (Local Prot Avail/In Use/Has BW/to NHOP)
      Label subobject: Flags 0x1, C-Type 1, Label 16
    10.1.1.2/32, Flags:0x0 (No Local Protection)
      Label subobject: Flags 0x1, C-Type 1, Label 0
  Resv ID handle: CD000404.
  Policy: Accepted. Policy source(s): MPLS/TE
```

Notice the following about the primary LSP:

- It has protection that uses a NHOP backup tunnel at its first hop.
- It has protection and is actively using an NHOP backup tunnel at its second hop.
- It has no local protection at its third hop.

The RRO display shows the following information for each hop:

- Whether local protection is available (that is, whether the LSP has selected a backup tunnel)
- Whether local protection is in use (that is, whether the LSP is currently using its selected backup tunnel)
- Whether the selected backup tunnel is an NHOP or NNHOP backup tunnel
- Whether the backup tunnel used at this hop provides bandwidth protection

Troubleshooting Tips

This section describes the following:

LSPs Do Not Become Active; They Remain Ready

At a PLR, LSPs transition from Ready to Active if one of the following events occurs:

- Primary interface goes down--If the primary interface (LSP’s outbound interface) goes down and the LSP is ready to use a backup tunnel, the LSP will transition to the active state causing its data to flow over the backup tunnel. On some platforms and interface types (for example, GSR POS interfaces), there is fast interface-down logic that detects this event very quickly. On other platforms where this logic does not exist, detection time is slower. On such platforms, it may be desirable to enable RSVP Hello (see the next bulleted item, “Hellos detect next hop is down”).
- Hellos detect next hop is down--If Hellos are enabled on the primary interface (LSP’s outbound interface), and the LSP’s next hop is no longer reachable, the next hop is declared down. This event will cause the LSP to begin actively using its backup tunnel. Notice that a next hop will be declared down even if the primary interface does not go down. For example, if the next hop stops responding due to a reboot or software orr hardware problem, Hellos will trigger the LSPs using this next hop to switch to their backup tunnels. Hellos can also help trigger FRR on interfaces such as Gigabit Ethernet where the interface remains up but is unusable (due to lack of link-layer liveness detection mechanisms).

Primary Tunnel Does Not Select Backup Tunnel That Is Up

If a backup tunnel is up, but it is not selected as a backup tunnel by the primary tunnel (LSP), enter the following commands for the backup tunnel:

- **shutdown**
- **no shutdown**

Note If you change the status of a backup tunnel, the backup tunnel selection algorithm is rerun for the backup tunnel. LSPs that have currently selected (that is, are ready to use) that backup tunnel will be disassociated from it, and then reassociated with that backup tunnel or another backup tunnel. This is generally harmless and usually results in mapping the same LSPs to that backup tunnel. However, if any LSPs are actively using that backup tunnel, shutting down the backup tunnel will tear down those LSPs.

Enhanced RSVP Commands Display Useful Information

The following RSVP commands have been enhanced to display information that can be helpful when you are examining the FRR state or troubleshooting FRR:

- **show ip rsvp request** --Displays upstream reservation state (that is, information related to the Resv messages that this node will send upstream).
- **show ip rsvp reservation** --Displays information about Resv messages received.
- **show ip rsvp sender** --Displays information about path messages being received.

These commands show control plane state; they do not show data state. That is, they show information about RSVP messages (Path and Resv) used to signal LSPs. For information about the data packets being forwarded along LSPs, use the **show mpls forwarding** command.

RSVP Hello Detects When a Neighboring Node Is Not Reachable

The RSVP Hello feature enables RSVP nodes to detect when a neighboring node is not reachable. Use this feature when notification of link-layer failures is not available and unnumbered links are not used, or when the failure detection mechanisms provided by the link layer are not sufficient for timely node failure detection. Hello must be configured both globally on the router and on the specific interface to be operational.

Hello Instances Have Not Been Created

If Hello instances have not been created, do the following:

- Determine if RSVP Hello has been enabled globally on the router. Enter the **ip rsvp signalling hello**(configuration) command.
- Determine if RSVP Hello has been enabled on an interface that the LSPs traverse. Enter the **ip rsvp signalling hello**(interface) command.
- Verify that at least one LSP has a backup tunnel by displaying the output of the **show ip rsvp sender** command. A value of “Ready” indicates that a backup tunnel has been selected.

“No entry at index” (error may self-correct, RRO may not yet have propagated from downstream node of interest)” Error Message Is Printed at the Point of Local Repair

FRR relies on a RRO in Resv messages arriving from downstream. Routers receiving path messages with the SESSION_ATTRIBUTE bit indicating that the LSP is fast-reroutable should include an RRO in the corresponding Resv messages.

If an LSP is configured for FRR, but the Resv arriving from a downstream router contains an incomplete RRO, the “No entry at index (error may self-correct, RRO may not yet have propagated from downstream node of interest)” message is printed. An incomplete RRO is one in which the NHOP or the NNHOP did not include an entry in the RRO.

This error typically means that backup tunnels to the NHOP or the NNHOP cannot be selected for this LSP because there is insufficient information about the NHOP or NNHOP due to the lack of an RRO entry.

Occasionally there are valid circumstances in which this situation occurs temporarily and the problem is self-corrected. If subsequent Resv messages arrive with a complete RRO, ignore the error message.

To determine whether the error has been corrected, display the RRO in Resv messages by entering the **clear ip rsvp hello instance counters** command. Use an output filter keyword to display only the LSP of interest.

“Couldn’t get rsbs” (error may self-correct when Resv arrives)” Error Message Is Printed at the Point of Local Repair

The PLR cannot select a backup tunnel for an LSP until a Resv message has arrived from downstream.

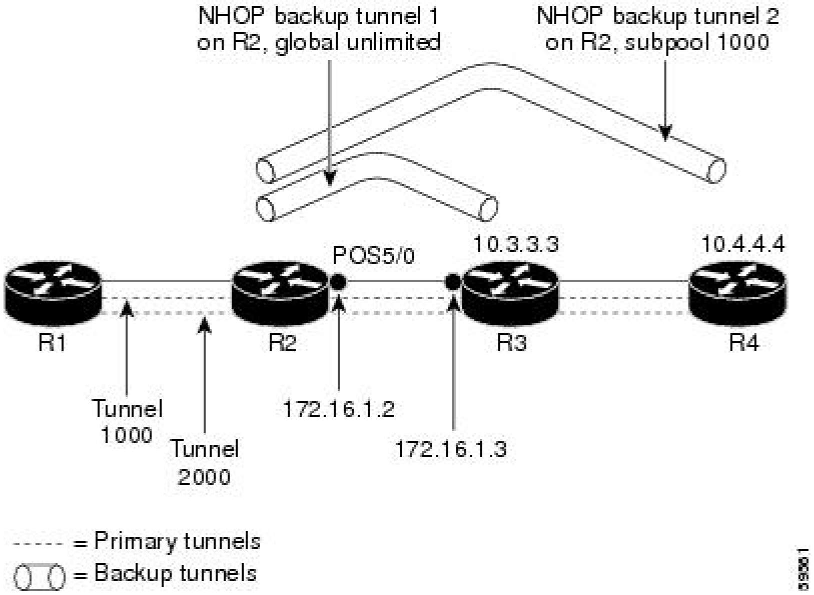
When this error occurs, it typically means that something is wrong. For example, no reservation exists for this LSP. You can troubleshoot this problem by using the **debug ip rsvp reservation** command to enable debug.

Occasionally there are valid circumstances in which this error message occurs and there is no need for concern. One such circumstance is when an LSP experiences a change before any Resv message has arrived from downstream. Changes can cause a PLR to try to select a backup tunnel for an LSP, and the selection will fail (causing this error message) if no Resv message has arrived for this LSP.

Configuration Examples for MPLS Traffic Engineering—Fast Reroute Link and Node Protection

The examples relate to the illustration shown in the figure below.

Figure 7: Backup Tunnels



Enabling Fast Reroute for all Tunnels Example

On router R1, enter interface configuration mode for each tunnel to be protected (Tunnel 1000 and Tunnel 2000). Enable these tunnels to use a backup tunnel in case of a link or node failure along their paths.

Tunnel 1000 will use 10 units of bandwidth from the subpool.

Tunnel 2000 will use five units of bandwidth from the global pool. The “bandwidth protection desired” bit has been set by specifying **bw-prot** in the **tunnel mpls traffic-eng fast-reroute** command.

```
Router(config)# interface Tunnel 1000
Router(config-if)# tunnel mpls traffic-eng fast-reroute
Router(config-if)# tunnel mpls traffic-eng bandwidth sub-pool 10
Router(config)# interface Tunnel2000
Router(config-if)# tunnel mpls traffic-eng fast-reroute bw-prot
Router(config-if)# tunnel mpls traffic-eng bandwidth 5
```

Configuring FRR on Port Channel

Note Ensure that the LACP minimum bundle configured is the same as number of member links.

In the example, port channel 1 has member 2 connected.

```
interface Port-channel1
 ip address 192.168.1.1 255.255.255.0
 negotiation auto
 mpls ip
 lacp min-bundle 2
```

Creating an NHOP Backup Tunnel Example

On router R2, create an NHOP backup tunnel to R3. This backup tunnel should avoid using the link 172.1.1.2.

```
Router(config)# ip explicit-path name avoid-protected-link
Router(cfg-ip-expl-path)# exclude-address 172.1.1.2
Explicit Path name avoid-protected-link:
____1: exclude-address 172.1.1.2
Router(cfg-ip_expl-path)# end
Router(config)# interface Tunnel 1
Router(config-if)# ip unnumbered loopback0
Router(config-if)# tunnel destination 10.3.3.3
Router(config-if)# tunnel mode mpls traffic-eng
Router(config-if)# tunnel mpls traffic-eng path-option 10 explicit avoid-protected-link
```

Creating an NNHOP Backup Tunnel Example

On router R2, create an NNHOP backup tunnel to R4. This backup tunnel should avoid R3.

```
Router(config)# ip explicit-path name avoid-protected-node

Router(cfg-ip-expl-path)# exclude-address 10.3.3.3
Explicit Path name avoid-protected-node:
____1: exclude-address 10.3.3.3
Router(cfg-ip_expl-path)# end

Router(config)# interface Tunnel 2

Router(config-if)# ip unnumbered loopback0

Router(config-if)# tunnel destination 10.4.4.4

Router(config-if)# tunnel mode mpls traffic-eng

Router(config-if)# tunnel mpls traffic-eng path-option 10 explicit avoid-protected-node
```

Assigning Backup Tunnels to a Protected Interface Example

On router R2, associate both backup tunnels with interface POS 5/0:

```
Router(config)# interface POS 5/0

Router(config-if)# mpls traffic-eng backup-path tunnel 1

Router(config-if)# mpls traffic-eng backup-path tunnel 2
```

Associating Backup Bandwidth and Pool Type with Backup Tunnels Example

Backup tunnel 1 is to be used only by LSPs that take their bandwidth from the global pool. It does not provide bandwidth protection. Backup tunnel 2 is to be used only by LSPs that take their bandwidth from the subpool. Backup tunnel 2 provides bandwidth protection for up to 1000 units.

```
Router(config)# interface Tunnel 1

Router(config-if)# tunnel mpls traffic-eng backup-bw global-pool Unlimited
Router(config)# interface Tunnel 2

Router(config-if)# tunnel mpls traffic-eng backup-bw sub-pool 1000
```

Configuring Backup Bandwidth Protection Example

In the following example, backup bandwidth protection is configured:

Note This global configuration is required only to change the backup protection preemption algorithm from minimize the number of LSPs that are demoted to minimize the amount of bandwidth that is wasted.

```
Router(config-if)# tunnel mpls traffic-eng fast-reroute bw-protect
Router(config)# mpls traffic-eng fast-reroute backup-prot-preemption optimize-bw
```

Configuring RSVP Hello and POS Signals Example

Hello must be configured both globally on the router and on the specific interface on which you need FRR protection. To configure Hello, use the following configuration commands:

- **ip rsvp signalling hello** (configuration)--Enables Hello globally on the router.
- **ip rsvp signalling hello** (interface)--Enables Hello on an interface where you need FRR protection.

The following configuration commands are optional:

- **ip rsvp signalling hello dscp** --Sets the differentiated services code point (DSCP) value that is in the IP header of the Hello message.
- **ip rsvp signalling hello refresh misses** --Specifies how many acknowledgments a node can miss in a row before the node considers that communication with its neighbor is down.

- **ip rsvp signalling hello refresh interval** --Configures the Hello request interval.
- **ip rsvp signalling hello statistics** --Enables Hello statistics on the router.

For configuration examples, see the Hello command descriptions in the “Command Reference” section of *MPLS Traffic Engineering (TE): Link and Node Protection, with RSVP Hellos Support* , Release 12.0(24)S.

To configure POS signaling for detecting FRR failures, enter the **pos report all** command or enter the following commands to request individual reports:

```
pos ais-shut
pos report rdool
pos report lais
pos report lrdi
pos report pais
pos report prdi
pos report sd-ber
```

CHAPTER 2

MPLS Traffic Engineering BFD-triggered Fast Reroute

The MPLS Traffic Engineering: BFD-triggered Fast Reroute feature allows you to obtain link and node protection by using the Bidirectional Forwarding Detection (BFD) protocol to provide fast forwarding path failure detection times for all media types, encapsulations, topologies, and routing protocols. In addition to fast forwarding path failure detection, BFD provides a consistent failure detection method for network administrators.

To obtain link and node protection by using the Resource Reservation Protocol (RSVP) with Hellos support, refer to the MPLS TE: Link and Node Protection, with RSVP Hellos Support (with Fast Tunnel Interface Down Detection) process module. RSVP Hellos enable a router to detect when a neighboring node has gone down but its interface to that neighbor is still operational.

Prerequisites for MPLS Traffic Engineering BFD-triggered Fast Reroute

- Configure BFD. Refer to the *Bidirectional Forwarding Detection* process module.
- Enable MPLS TE on all relevant routers and interfaces.
- Configure MPLS TE tunnels.
- For additional prerequisites, refer to the MPLS TE: Link and Node Protection, with RSVP Hellos Support (with Fast Tunnel Interface Down Detection) process module.

Restrictions for MPLS Traffic Engineering BFD-triggered Fast Reroute

- You cannot configure BFD and RSVP Hellos on the same interface.
- BFD may not be supported on some interfaces.
- For additional restrictions, refer to the MPLS TE: Link and Node Protection, with RSVP Hellos Support (with Fast Tunnel Interface Down Detection) process module.

Information About MPLS Traffic Engineering BFD-triggered Fast Reroute

Bidirectional Forwarding Detection

Bidirectional Forwarding Detection (BFD) is a detection protocol designed to provide fast forwarding path failure detection times for all media types, encapsulations, topologies, and routing protocols. In addition to fast forwarding path failure detection, BFD provides a consistent failure detection method for network administrators. Because the network administrator can use BFD to detect forwarding path failures at a uniform rate, rather than the variable rates for different routing protocol Hello mechanisms, network profiling and planning will be easier, and reconvergence time will be consistent and predictable.

Fast Reroute

Fast Reroute (FRR) is a mechanism for protecting Multiprotocol Label Switching (MPLS) traffic engineering (TE) label switched paths (LSPs) from link and node failures by locally repairing the LSPs at the point of failure, allowing data to continue to flow on them while their headend routers attempt to establish new end-to-end LSPs to replace them. FRR locally repairs the protected LSPs by rerouting them over backup tunnels that bypass failed links or nodes.

Link Protection

Backup tunnels that bypass only a single link of the LSP’s path provide link protection. They protect LSPs if a link along their path fails by rerouting the LSP’s traffic to the next hop (bypassing the failed link). These are referred to as next-hop (NHOP) backup tunnels because they terminate at the LSP’s next hop beyond the point of failure.

Node Protection

FRR provides node protection for LSPs. Backup tunnels that bypass next-hop nodes along LSP paths are called next-next-hop (NNHOP) backup tunnels because they terminate at the node following the next-hop node of the LSP paths, thereby bypassing the next-hop node. They protect LSPs if a node along their path fails by enabling the node upstream of the failure to reroute the LSPs and their traffic around the failed node to the next-next hop. FRR supports the use of RSVP Hellos to accelerate the detection of node failures. NNHOP backup tunnels also provide protection from link failures, because they bypass the failed link as well as the node.

Bandwidth Protection

NHOP and NNHOP backup tunnels can be used to provide bandwidth protection for rerouted LSPs. This is referred to as backup bandwidth. You can associate backup bandwidth with NHOP or NNHOP backup tunnels. This informs the router of the amount of backup bandwidth a particular backup tunnel can protect. When a router maps LSPs to backup tunnels, bandwidth protection ensures that an LSP uses a given backup tunnel only if there is sufficient backup bandwidth. The router selects which LSPs use which backup tunnels to provide maximum bandwidth protection. That is, the router determines the best way to map LSPs onto backup tunnels to maximize the number of LSPs that can be protected. .

LSPs that have the “bandwidth protection desired” bit set have a higher right to select backup tunnels that provide bandwidth protection; that is, those LSPs can preempt other LSPs that do not have that bit set. For more information, see the "Prioritizing Which LSPs Obtain Backup Tunnels with Bandwidth Protection" section.

How to Configure MPLS Traffic Engineering BFD-triggered Fast Reroute

This section shows you how to add FRR protection to a network in which MPLS TE LSPs are configured.

The following sections describe how to use FRR to protect LSPs in your network from link or node failures. Each task is identified as either required or optional.

Note You can perform the configuration tasks in any order.

Note An NNHOP backup tunnel must *not* go via the NHOP backup tunnel.

Enabling BFD Support on the Router

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip rsvp signalling hello bfd**
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip rsvp signalling hello bfd** **Example:** `Router(config)# ip rsvp signalling hello bfd`	Enables the BFD protocol on the router for MPLS TE link and node protection.
Step 4	**exit** **Example:** `Router(config)# exit`	Exits to privileged EXEC mode.

Enabling Fast Reroute on LSPs

LSPs can use backup tunnels only if the LSPs have been configured as fast reroutable. To enable FRR on the LSP, enter the following commands at the headend of each LSP.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *number*
4. **tunnel mpls traffic-eng fast-reroute** [**bw-protect**] [**node-protect**]
5. **exit**
6. **exit**

DETAILED STEPS

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

	Command or Action	Purpose
	`Router# configure terminal`	
Step 3	**interface tunnel** *number* **Example:** `Router(config)# interface tunnel 1000`	Enters interface configuration mode for the specified tunnel. • The *number* argument is the number of the tunnel.
Step 4	**tunnel mpls traffic-eng fast-reroute** [**bw-protect**] [**node-protect**] **Example:** `Router(config-if)# tunnel mpls traffic-eng fast-reroute bw-protect node-protect`	Enables an MPLS TE tunnel to use an established backup tunnel if there is a link or node failure. • The **bw-protect** keyword sets the "bandwidth protection desired" bit so that backup bandwidth protection is enabled. • The **node-protect** keyword sets the "node protection desired" bit so that backup bandwidth protection is enabled.
Step 5	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode and returns to global configuration mode.
Step 6	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode and returns to privileged EXEC mode.

Creating a Backup Tunnel to the Next Hop or to the Next-Next Hop

To create a backup tunnel to the next hop or to the next-next hop, perform the following task.

Enter the commands on the node that will be the headend of the backup tunnel (that is, the node whose downstream link or node may fail). The node on which you enter the commands must be a supported platform. See the Finding Feature Information section.

Creating a backup tunnel is basically no different from creating any other tunnel.

Note When using the **exclude-address** command to specify the path for a backup tunnel, you must exclude an interface address to avoid a link (for creating an NHOP backup tunnel), or a router-ID address to avoid a node (for creating an NNHOP backup tunnel).

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **interface tunnel** *number*
4. **ip unnumbered** *type number*
5. **tunnel destination** *ip-address*
6. **tunnel mode mpls traffic-eng**
7. **tunnel mpls traffic-eng path-option** *number* {**dynamic** | **explicit** {**name** *path-name* | *path-number*}}[**lockdown**]
8. **exit**
9. **ip explicit-path name** *name*
10. **exclude-address** *address*
11. **exit**
12. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface tunnel** *number* **Example:** `Router(config)# interface tunnel 1`	Creates a new tunnel interface and enters interface configuration mode. • The *number* argument is the number of the tunnel.
Step 4	**ip unnumbered** *type number* **Example:** `Router(config-if)# ip unnumbered loopback 0`	Enables IP processing on an interface without assigning an explicit IP address to the interface. • The *type* and *number* arguments name the type and number of another interface on which the router has an assigned IP address. It cannot be another unnumbered interface. **Note** The **ip unnumbered loopback 0** command gives the tunnel interface an IP address that is the same as that of interface loopback 0. This command is not effective until loopback 0 has been configured with an IP address.
Step 5	**tunnel destination** *ip-address* **Example:** `Router(config-if)# tunnel destination 10.3.3.3`	Specifies the destination for a tunnel interface. • The *ip-address*argument specifies the IP address of the device, expressed in dotted decimal notation, where the tunnel will terminate. That address should

	Command or Action	Purpose
		be the router ID of the device that is the NHOP or NNHOP of LSPs to be protected.
Step 6	**tunnel mode mpls traffic-eng** **Example:** `Router(config-if)# tunnel mode mpls traffic-eng`	Sets encapsulation mode of the tunnel to MPLS TE.
Step 7	**tunnel mpls traffic-eng path-option** *number* {**dynamic** | **explicit** {**name** *path-name* | *path-number*}}[**lockdown**] **Example:** `Router(config-if)# tunnel mpls traffic-eng path-option 10 explicit name avoid-protected-link`	Configures the tunnel to use a named IP explicit path or a path dynamically calculated from the traffic engineering topology database. • The *number* argument is the preference for this path option. When you configure multiple path options, lower numbered options are preferred. Valid values are from 1 to 1000. • The **dynamic** keyword indicates that the path of the label switched path (LSP) is dynamically calculated. • The **explicit** keyword indicates that the path of the LSP is an IP explicit path. • The **name** *path-name* keyword and argument are the path name of the IP explicit path that the tunnel uses with this option. • The **identifier** *path-number*keyword and argument pair names the path number of the IP explicit path that the tunnel uses with this option. The range is from 1 to 65535. • The **lockdown** keyword specifies that The LSP cannot be reoptimized. **Note** A dynamic path is used if an explicit path is currently unavailable.
Step 8	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode and enter global configuration mode.
Step 9	**ip explicit-path name** *name* **Example:** `Router(config)# ip explicit-path name avoid-protected-link`	Enters IP explicit path mode for IP explicit paths to create the named path. • The *name* argument is the name of the explicit path.
Step 10	**exclude-address** *address*	Excludes an address from an explicit-path.

	Command or Action	Purpose
	Example: `Router(cfg-ip-expl-path)# exclude-address 10.3.3.3`	- The *address* argument specifies the IP address of the link to be protected for link protection. For node protection, it specifies the router ID of the node to be protected. **Note** Backup tunnel paths can be dynamic or explicit and they do not have to use an excluded address. Because backup tunnels must avoid the protected link or node, it is convenient to use an excluded address.
Step 11	**exit** **Example:** `Router(cfg-ip-expl-path))# exit`	Exits IP explicit path configuration mode and returns to global configuration mode.
Step 12	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode and returns to privileged EXEC mode.

Assigning Backup Tunnels to a Protected Interface

To assign one or more backup tunnels to a protected interface, perform the following task.

Enter the commands on the node that will be the headend of the backup tunnel (that is, the node whose downstream link or node may fail). The node on which you enter the commands must be a supported platform. See the Finding Feature Information section.

Note You must configure the interface to have an IP address and to enable the MPLS TE tunnel feature.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type* slot/sub*slot* **/** *port*[**.** *subinterface*]
4. **mpls traffic-eng backup-path tunnel** *tunnel-id*
5. **exit**
6. **exit**

DETAILED STEPS

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

	Command or Action	Purpose
	Example: `Router> enable`	- Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type* slot/sub*slot* / *port*[**.** *subinterface*] **Example:** `Router(config)# interface Gigabitethernet 2/1/0`	Configures an interface type and enters interface configuration mode. - The *type* argument is the type of interface to be configured. - The *slot* argument is the chassis slot number. Refer to the appropriate hardware manual for slot information. For SIPs, refer to the platform-specific SPA hardware installation guide or the corresponding “Identifying Slots and Subslots for SIPs and SPAs” topic in the platform-specific SPA software configuration guide. - The / *subslot* keyword and argument pair is the secondary slot number on a SIP where a SPA is installed. The slash (/) is required. Refer to the platform-specific SPA hardware installation guide and the corresponding “Specifying the Interface Address on a SPA” topic in the platform-specific SPA software configuration guide for subslot information. - The / *port* keyword and argument pair is the port or interface number. The slash (/) is required. Refer to the appropriate hardware manual for port information. For SPAs, refer to the corresponding “Specifying the Interface Address on a SPA” topics in the platform-specific SPA software configuration guide - The **.** *subinterface-number*keyword and argument pair is the subinterface number in the range 1 to 4294967293. The number that precedes the period (.) must match the number to which this subinterface belongs.
Step 4	**mpls traffic-eng backup-path tunnel** *tunnel-id* **Example:** `Router(config-if)# mpls traffic-eng backup-path tunnel2`	Configures the physical interface to use for a backup tunnel in the event of a detected failure on that interface. - The *tunnel-id*argument is a string that identifies a backup tunnel to use if there is a link or node failure for LSPs going out the configured interface.

	Command or Action	Purpose
		Note You can enter this command multiple times to associate multiple backup tunnels with the same protected interface.
Step 5	**exit** **Example:** `Router(config-if))# exit`	Exits interface configuration mode and returns to global configuration mode.
Step 6	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode and returns to privileged EXEC mode.

Enabling BFD on the Protected Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type* slot/sub*slot* / *port*[**.** *subinterface*]
4. **ip rsvp signalling hello bfd**
5. **bfd interval** *milliseconds* **min_rx** *milliseconds* **multiplier** *interval-multiplier*
6. **exit**
7. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type* slot/sub*slot* / *port*[**.** *subinterface*] **Example:** `Router(config)# interface Gigabitethernet 2/1/0`	Configures an interface type and enters interface configuration mode. • The *type* argument is the type of interface to be configured.

	Command or Action	Purpose
		• The *slot* argument is the chassis slot number. Refer to the appropriate hardware manual for slot information. For SIPs, refer to the platform-specific SPA hardware installation guide or the corresponding “Identifying Slots and Subslots for SIPs and SPAs” topic in the platform-specific SPA software configuration guide. • The / *subslot* keyword and argument pair is the secondary slot number on a SIP where a SPA is installed. The slash (/) is required. Refer to the platform-specific SPA hardware installation guide and the corresponding “Specifying the Interface Address on a SPA” topic in the platform-specific SPA software configuration guide for subslot information. • The / *port* keyword and argument pair is the port or interface number. The slash (/) is required. Refer to the appropriate hardware manual for port information. For SPAs, refer to the corresponding “Specifying the Interface Address on a SPA” topics in the platform-specific SPA software configuration guide • The **.** *subinterface-number*keyword and argument pair is the subinterface number in the range 1 to 4294967293. The number that precedes the period (.) must match the number to which this subinterface belongs.
Step 4	**ip rsvp signalling hello bfd** **Example:** `Router(config-if)# ip rsvp signalling hello bfd`	Enables the BFD protocol on an interface for MPLS TE link and node protection.
Step 5	**bfd interval** *milliseconds* **min_rx** *milliseconds* **multiplier** *interval-multiplier* **Example:** `Router(config-if)# bfd interval 100 min_rx 100 multiplier 4`	Sets the BFD session parameters for an interface. • The **interval** *milliseconds* keyword and argument pair specifies the rate at which BFD control packets will be sent to BFD peers. The configurable time period for the milliseconds argument is from 50 to 999. • The **min_rx** *millisecond* keyword and argument pair specifies the rate at which BFD control packets will be expected to be received from BFD peers. The configurable time period for the milliseconds argument is from 1 to 999. • The **multiplier** i*nterval-multiplier*keyword and argument pair specifies the number of consecutive BFD control packets that must be missed from a BFD peer before BFD declares that the peer is unavailable

	Command or Action	Purpose
		and the Layer 3 BFD peer is informed of the failure. The configurable value range for the multiplier-value argument is from 3 to 50.
Step 6	**exit** **Example:** `Router(config-if))# exit`	Exits interface configuration mode and returns to global configuration mode.
Step 7	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode and returns to privileged EXEC mode.

Associating Backup Bandwidth and Pool Type with a Backup Tunnel

To associate backup bandwidth with a backup tunnel and designate the type of LSP that can use a backup tunnel, enter the following tasks.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *number*
4. **tunnel mpls traffic-eng backup-bw** {*bandwidth* | [**sub-pool** {*bandwidth* | **Unlimited**}] [**global-pool** {*bandwidth* | **Unlimited**}]} [**any** {*bandwidth* | **Unlimited**}]
5. **exit**
6. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface tunnel** *number* **Example:** `Router(config)# interface tunnel 2`	Enters interface configuration mode for the specified tunnel. • The *number* argument is the number of the tunnel.

	Command or Action	Purpose
Step 4	**tunnel mpls traffic-eng backup-bw** {*bandwidth* | [**sub-pool** {*bandwidth* | **Unlimited**}] [**global-pool** {*bandwidth* | **Unlimited**}]} [**any** {*bandwidth* | **Unlimited**}] **Example:** `Router(config-if)# tunnel mpls traffic-eng backup-bw sub-pool 1000`	Associates bandwidth with a backup tunnel and designates whether LSPs that allocate bandwidth from the specified pool can use the tunnel.
Step 5	**exit** **Example:** `Router(config-if))# exit`	Exits interface configuration mode and returns to global configuration mode.
Step 6	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode and returns to privileged EXEC mode.

Configuring Backup Bandwidth Protection

To configure the backup bandwidth protection, perform the following task.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *number*
4. **tunnel mpls traffic-eng fast-reroute** [**bw-protect**]
5. **exit**
6. **mpls traffic-eng fast-reroute backup-prot-preemption** [**optimize-bw**]
7. **exit**

DETAILED STEPS

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

	Command or Action	Purpose
Step 3	**interface tunnel** *number* **Example:** `Router(config)# interface tunnel 2`	Enters interface configuration mode for the specified tunnel.
Step 4	**tunnel mpls traffic-eng fast-reroute** [**bw-protect**] **Example:** `Router(config-if)# tunnel mpls traffic-eng fast-reroute bw-protect`	Enables an MPLS TE tunnel to use an established backup tunnel in the event of a link or node failure. • The **bw-protect** keyword gives an LSP priority for using backup tunnels with bandwidth protection.
Step 5	**exit** **Example:** `Router(config-if)# exit`	Exits to global configuration mode.
Step 6	**mpls traffic-eng fast-reroute backup-prot-preemption** [**optimize-bw**] **Example:** `Router(config)# mpls traffic-eng fast-reroute backup-prot-preemption optimize-bw`	Changes the backup protection preemption algorithm from minimize the number of LSPs that are demoted to minimize the amount of bandwidth that is wasted.
Step 7	**exit** **Example:** `Router(config-if)# exit`	Exits to privileged EXEC mode.

Verifying That Fast Reroute Is Operational

SUMMARY STEPS

1. **show mpls traffic-eng tunnels brief**
2. **show ip rsvp sender detail**
3. **show mpls traffic-eng fast-reroute database**
4. **show mpls traffic-eng tunnels backup**
5. **show mpls traffic-eng fast-reroute database**
6. **show ip rsvp reservation detail**
7. **show ip rsvp hello**
8. **show ip rsvp interface detail**
9. **show ip rsvp hello bfd nbr**
10. **show ip rsvp hello bfd nbr detail**
11. **show ip rsvp hello bfd nbr summary**

DETAILED STEPS

Step 1 **show mpls traffic-eng tunnels brief**

Use this command to verify that backup tunnels are up:

Example:

```
Router# show mpls traffic-eng tunnels brief

Signalling Summary:
    LSP Tunnels Process:           running
    RSVP Process:                  running
    Forwarding:                    enabled
    Periodic reoptimization:       every 3600 seconds, next in 1706 seconds
TUNNEL NAME                      DESTINATION      UP IF     DOWN IF   STATE/PROT
Router_t1                        10.112.0.12      -         Gi4/0/1   up/up
Router_t2                        10.112.0.12      -         unknown   up/down
Router_t3                        10.112.0.12      -         unknown   admin-down
Router_t1000                     10.110.0.10      -         unknown   up/down
Router_t2000                     10.110.0.10      -         Gi4/0/1   up/up
Displayed 5 (of 5) heads, 0 (of 0) midpoints, 0 (of 0) tails
```

Step 2 **show ip rsvp sender detail**

Use this command to verify that LSPs are protected by the appropriate backup tunnels.

Following is sample output from the **show ip rsvp sender detail** command when the command is entered at the router acting as the point of local repair (PLR) before a failure:

Example:

```
Router# show ip rsvp sender detail

PATH:
 Tun Dest:   10.10.0.6  Tun ID: 100  Ext Tun ID: 10.10.0.1
 Tun Sender: 10.10.0.1  LSP ID: 31
 Path refreshes:
  arriving: from PHOP 10.10.7.1 on Et0/0 every 30000 msecs
 Session Attr:
  Setup Prio: 7, Holding Prio: 7
  Flags: (0x7) Local Prot desired, Label Recording, SE Style
  session Name: R1_t100
 ERO: (incoming)
  10.10.7.2 (Strict IPv4 Prefix, 8 bytes, /32)
  10.10.0.6 (Strict IPv4 Prefix, 8 bytes, /32)
 RRO:
   10.10.7.1/32, Flags:0x0 (No Local Protection)
   10.10.4.1/32, Flags:0x9 (Local Prot Avail/to NNHOP) !Available to NNHOP
   10.10.1.1/32, Flags:0x0 (No Local Protection)
 Traffic params - Rate: 10K bits/sec, Max. burst: 1K bytes
   Min Policed Unit: 0 bytes, Max Pkt Size 4294967295 bytes
 Fast-Reroute Backup info:
   Inbound  FRR: Not active
   Outbound FRR: No backup tunnel selected
 Path ID handle: 50000416.
 Incoming policy: Accepted. Policy source(s): MPLS/TE
 Status: Proxy-terminated
```

Step 3 **show mpls traffic-eng fast-reroute database**

Enter the **clear ip rsvp hello instance counters**command to verify the following:

- MPLS TE FRR Node Protection has been enabled.
- A certain type of LSP can use a backup tunnel.

The following command output displays the LSPs that are protected:

Example:

```
Router# show mpls traffic-eng fast-reroute database
Tunnel head end item frr information:
Protected tunnel              In-label Out intf/label   FRR intf/label   Status
Tunnel500                     Tun hd   AT4/0.100:Untagg Tu501:20         ready
Prefix item frr information:
Prefix            Tunnel      In-label Out intf/label    FRR intf/label  Status
10.0.0.8/32       Tu500       18       AT4/0.100:Pop ta  Tu501:20        ready
10.0.8.8/32       Tu500       19       AT4/0.100:Untagg  Tu501:20        ready
10.8.9.0/24       Tu500       22       AT4/0.100:Untagg  Tu501:20        ready
LSP midpoint item frr information:
LSP identifier     In-label Out     intf/label     FRR intf/label     Status
```

If Label Distribution Protocol (LDP) is not enabled, separate prefix items are not shown because all prefixes then use a single rewrite. To confirm that a particular IP prefix is FRR protected, even though it is not shown in this display, enter it within the **show mpls forwarding-table** *ip-address* **detail** command. The final line of the display will tell whether that prefix is protected:

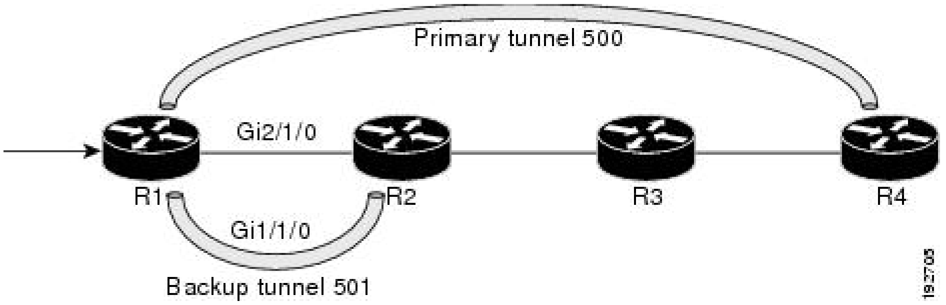
Example:

```
Router# show mpls forwarding-table 10.0.0.11 32 detail

Local    Outgoing    Prefix         Bytes tag   Outgoing               Next Hop
tag      tag or VC   or Tunnel Id   switched    interface
Tun hd   Untagged    10.0.0.11/32   48 5/0      Gi5/0    point2point
          MAC/Encaps=4/8, MTU=1520, Tag Stack{22}
          48D18847 00016000
          No output feature configured
          Fast Reroute Protection via (Tu0, outgoing label 12304)
```

The following command output displays the LSPs that are protected when the FRR primary tunnel is over a Gigabit Ethernet interface and the backup tunnel is over a Gigabit Ethernet interface. As shown in the figure below, interface Gigabit Ethernet 2/1/0 is protected by backup tunnel 501.

Figure 8: Protected LSPs



The figure above shows the following:

- Primary tunnel 500--Path is R1 via Gigabit Ethernet2/1/0 to R2 to R3 to R4.

- FRR backup tunnel 501--Path is R1 via Gigabit Ethernet1/1/0 to R2.
- Interface Gigabit Ethernet1/1/0--Protected by backup tunnel 501.

Example:

```
Router# show mpls traffic-eng fast-reroute database
Tunnel head end item frr information:
Protected tunnel In-label Out intf/label FRR intf/label Status
Tunnel500 Tun hd AT4/0.100:Untagg Tu501:20 ready
Prefix item frr information:
Prefix Tunnel In-label Out intf/label FRR intf/label Status
10.0.0.8/32 Tu500 18 AT4/0.100:Pop ta Tu501:20 ready
10.0.8.8/32 Tu500 19 AT4/0.100:Untagg Tu501:20 ready
10.8.9.0/24 Tu500 22 AT4/0.100:Untagg Tu501:20 ready
LSP midpoint item frr information:
LSP identifier In-label Out intf/label FRR intf/label Status
```

The following command output displays the LSPs that are protected when the FRR backup tunnel is over a Gigabit Ethernet interface.

Example:

```
Router# show mpls traffic-eng fast-reroute database

Tunnel head end item frr information:
Protected tunnel In-label Out intf/label FRR intf/label Status
Tunnel500 Tun hd PO2/0:Untagged Tu501:20 ready
Prefix item frr information:
Prefix Tunnel In-label Out intf/label FRR intf/label Status
10.0.0.8/32 Tu500 18 PO2/0:Pop tag Tu501:20 ready
10.0.8.8/32 Tu500 19 PO2/0:Untagged Tu501:20 ready
10.8.9.0/24 Tu500 22 PO2/0:Untagged Tu501:20 ready
LSP midpoint item frr information:
LSP identifier In-label Out intf/label FRR intf/label Status
```

Step 4 **show mpls traffic-eng tunnels backup**

For backup tunnels to be operational, the LSP must be reroutable. At the headend of the LSP, enter the **show run interface tunnel** *tunnel-number* command. The output should include the **tunnel mpls traffic-eng fast-reroute** command. If it does not, enter this command for the tunnel.

On the router where the backup tunnels originate, enter the **show mpls traffic-eng tunnels backup**command. Following is sample command output:

Example:

```
Router# show mpls traffic-eng tunnels backup
Router_t578
  LSP Head, Tunnel578, Admin: up, Oper: up
  Src 10.55.55.55, Dest 10.88.88.88, Instance 1
  Fast Reroute Backup Provided:
    Protected i/fs: PO1/0, PO1/1, PO3/3
    Protected lsps: 1
    Backup BW: any pool unlimited; inuse: 100 kbps
Router_t5710
  LSP Head, Tunnel5710, Admin: admin-down, Oper: down
  Src 10.55.55.55, Dest 10.7.7.7, Instance 0
  Fast Reroute Backup Provided:
    Protected i/fs: PO1/1
    Protected lsps: 0
    Backup BW: any pool unlimited; inuse: 0 kbps
```

```
Router_t5711
  LSP Head, Tunnel5711, Admin: up, Oper: up
  Src 10.55.55.55, Dest 10.7.7.7, Instance 1
  Fast Reroute Backup Provided:
    Protected i/fs: PO1/0
    Protected lsps: 2
    Backup BW: any pool unlimited; inuse: 6010 kbps
```

The command output will allow you to verify the following:

- Backup tunnel exists--Verify that there is a backup tunnel that terminates at this LSP’s NHOP or NNHOP. Look for the LSP’s NHOP or NNHOP in the Dest field.
- Backup tunnel is up--To verify that the backup tunnel is up, look for “Up” in the Oper field.
- Backup tunnel is associated with the LSP’s interface--Verify that the interface for the LSP is allowed to use this backup tunnel. Look for the LSP’s output interface in the protected i/fs field list.
- Backup tunnel has sufficient bandwidth--If you restricted the amount of bandwidth a backup tunnel can hold, verify that the backup tunnel has sufficient bandwidth to hold the LSPs that would use this backup tunnel if there is a failure. The bandwidth of an LSP is defined by the line **tunnel mpls traffic-eng bandwidth** at the headend of the LSP. To determine the available bandwidth on a backup tunnel, look at the “cfg” and “inuse” fields. If there is insufficient backup bandwidth to accommodate the LSPs that would use this backup tunnel in the event of a failure, create an additional backup tunnel or increase the backup bandwidth of the existing tunnel by using the **tunnel mpls traffic-eng bandwidth** command.

Note In order to determine how much bandwidth is sufficient, offline capacity planning may be required.

Backup tunnel has appropriate bandwidth type--If you restricted the type of LSPs (subpool or global pool) that can use this backup tunnel, verify that the LSP is the appropriate type for the backup tunnel. The type of the LSP is defined by the line **tunnel mpls traffic-eng bandwidth** at the headend of this LSP. If this line contains the word “sub pool”, then it uses subpool bandwidth; otherwise, it uses global pool bandwidth. Verify that the type matches the type the backup tunnel can hold by looking in the output of the **tunnel mpls traffic-eng bandwidth**command.

If none of the verification actions described succeed, enable debug by entering the **debug ip rsvp fast-reroute** command and the **debug mpls traffic-eng fast-reroute** command on the router that is the headend of the backup tunnel. Then do the following:

a. Enter the **shutdown** command for the primary tunnel.

b. Enter the **no shutdown** command for the primary tunnel.

c. View the debug output.

Step 5 **show mpls traffic-eng fast-reroute database**

Enter the **clear ip rsvp hello instance counters** command to verify the following:

- MPLS TE FRR node protection has been enabled.
- A certain type of LSP can use a backup tunnel.

The following command output displays the LSPs that are protected:

Example:

```
Router# show mpls traffic-eng fast-reroute database
```

```
Tunnel head end item frr information:
Protected Tunnel   In-label   intf/label       FRR intf/label      Status
Tunnel10           Tun        Gi0/1/0:Untagged   Tu0:12304           ready
Prefix item frr information:
Prefix        Tunnel  In-label    Out intf/label  FRR intf/label   Status
10.0.0.11/32  Tu110   Tun hd      Gi0/1/0:Untagged  Tu0:12304        ready
LSP midpoint frr information:
LSP identifier       In-label   Out intf/label   FRR intf/label    Status
10.0.0.12 1 [459]    16         Gi0/1/1:17         Tu2000:19         ready
```

Note If Label Distribution Protocol (LDP) is not enabled, separate prefix items are not shown because all prefixes then use a single rewrite. To confirm that a particular IP prefix is FRR protected, even though it is not shown in this display, enter it within the **show mpls forwarding-table** *ip-address* **detail** command. The final line of the display will tell whether that prefix is protected.

Example:

```
Router# show mpls forwarding-table 10.0.0.11 32 detail

Local    Outgoing    Prefix         Bytes tag    Outgoing       Next Hop
tag      tag or VC   or Tunnel Id   switched     interface
Tun hd   Untagged    10.0.0.11/32   48 Gi0/1/0     point2point
          MAC/Encaps=4/8, MTU=1520, Tag Stack{22}
          48D18847 00016000
          No output feature configured
          Fast Reroute Protection via (Tu0, outgoing label 12304)
```

Step 6 **show ip rsvp reservation detail**

Following is sample output from the **show ip rsvp reservation detail** command entered at the headend of a primary LSP. Entering the command at the headend of the primary LSP shows, among other things, the status of FRR (that is, local protection) at each hop this LSP traverses. The per-hop information is collected in the Record Route Object (RRO) that travels with the Resv message from the tail to the head.

Example:

```
Router# show ip rsvp reservation detail
Reservation:
  Tun Dest:   10.1.1.1  Tun ID: 1  Ext Tun ID: 10.1.1.1
  Tun Sender: 10.1.1.1  LSP ID: 104
  Next Hop: 10.1.1.2 on Gi1/0/2
  Label: 18 (outgoing)
  Reservation Style is Shared-Explicit, QoS Service is Controlled-Load
  Average Bitrate is 0 bits/sec, Maximum Burst is 1K bytes
  Min Policed Unit: 0 bytes, Max Pkt Size: 0 bytes
  RRO:
    10.1.1.1/32, Flags:0x1 (Local Prot Avail/to NHOP)
      Label subobject: Flags 0x1, C-Type 1, Label 18
    10.1.1.1/32, Flags:0x0 (Local Prot Avail/In Use/Has BW/to NHOP)
      Label subobject: Flags 0x1, C-Type 1, Label 16
    10.1.1.2/32, Flags:0x0 (No Local Protection)
      Label subobject: Flags 0x1, C-Type 1, Label 0
  Resv ID handle: CD000404.
  Policy: Accepted. Policy source(s): MPLS/TE
```

Notice the following about the primary LSP:

- It has protection that uses an NHOP backup tunnel at its first hop.
- It has protection and is actively using an NHOP backup tunnel at its second hop.
- It has no local protection at its third hop.

The RRO display shows the following information for each hop:

- Whether local protection is available (that is, whether the LSP has selected a backup tunnel)
- Whether local protection is in use (that is, whether the LSP is using its selected backup tunnel)
- Whether the selected backup tunnel is an NHOP or NNHOP backup tunnel
- Whether the backup tunnel used at this hop provides bandwidth protection

Step 7 **show ip rsvp hello**

Use this command to display hello status and statistics for FRR, reroute (hello state timer), and graceful restart. Following is sample output:

Example:

```
Router# show ip rsvp hello

Hello:
 RSVP Hello for Fast-Reroute/Reroute: Enabled
  Statistics: Disabled
 BFD for Fast-Reroute/Reroute: Enabled
 RSVP Hello for Graceful Restart: Disabled
```

Step 8 **show ip rsvp interface detail**

Use this command to display the interface configuration for Hello. Following is sample output:

Example:

```
Router# show ip rsvp interface detail

Gi2/1/1:
 RSVP: Enabled
 Interface State: Up
 Bandwidth:
  Curr allocated: 0 bits/sec
  Max. allowed (total): 0 bits/sec
  Max. allowed (per flow): 0 bits/sec
  Max. allowed for LSP tunnels using sub-pools (pool 1): 0 bits/sec
  Set aside by policy (total): 0 bits/sec
 Signalling:
  DSCP value used in RSVP msgs: 0x3F
  Number of refresh intervals to enforce blockade state: 4
 Authentication: disabled
  Key chain: <none>
  Type:  md5
  Window size: 1
  Challenge:  disabled
 FRR Extension:
  Backup Path: Configured (or "Not Configured")
 BFD Extension:
  State: Disabled
  Interval: Not Configured
 RSVP Hello Extension:
  State: Disabled
  Refresh Interval: FRR: 200  , Reroute: 2000
  Missed Acks:      FRR: 4    , Reroute: 4
  DSCP in HELLOs:   FRR: 0x30 , Reroute: 0x30
```

Step 9 **show ip rsvp hello bfd nbr**

Use this command to display information about all MPLS traffic engineering link and node protected neighbors that use the BFD protocol. Following is sample output. The command output is the same as the **show ip rsvp hello bfd nbr summary** command output.

Example:

```
Router# show ip rsvp hello bfd nbr

Client  Neighbor    I/F     State   LostCnt   LSPs
FRR     10.0.0.6    Gi2/1/1  Up      0         1
```

Step 10 **show ip rsvp hello bfd nbr detail**

Use this command to display detailed information about all MPLS traffic engineering link and node protected neighbors that use the BFD protocol:

Example:

```
Router# show ip rsvp hello bfd nbr detail

 Hello Client Neighbors
 Remote addr 10.0.0.6, Local addr  10.0.0.7
  Type: Active
  I/F: Gi2/1/1
  State: Up (for 00:09:41)
  Clients: FRR
  LSPs protecting: 1 (frr: 1, hst upstream: 0 hst downstream: 0)
  Communication with neighbor lost: 0
```

Step 11 **show ip rsvp hello bfd nbr summary**

Use this command to display summarized information about all MPLS traffic engineering link and node protected neighbors that use the BFD protocol. The command output is the same as the **show ip rsvp hello bfd nbr summary** command output.

Example:

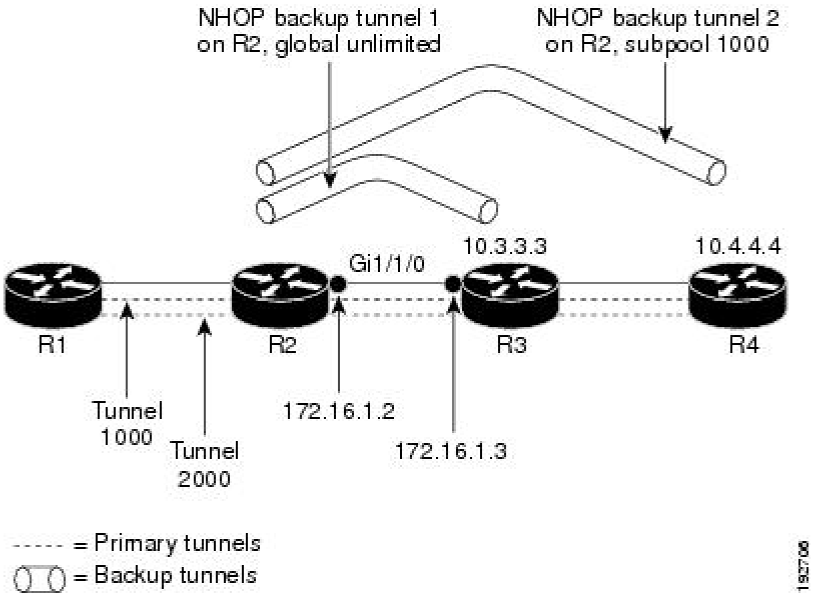
```
Router# show ip rsvp hello bfd nbr summary

Client  Neighbor    I/F     State  LostCnt  LSPs
FRR     10.0.0.6   Gi2/1/1  Up     0        1
```

Configuration Examples for MPLS Traffic Engineering BFD-triggered Fast Reroute

The examples in this section are based on the backup tunnels shown in the figure below.

Figure 9: Backup Tunnels



Example Enabling BFD Support on the Router

The following example enables the BFD protocol on the router:

```
Router(config)# ip rsvp signalling hello bfd
```

Example Enabling Fast Reroute on LSPs

On router R1 in the figure above, enter interface configuration mode for each tunnel to be protected (Tunnel 1000 and Tunnel 2000). Enable these tunnels to use a backup tunnel in case of a link or node failure along their paths.

Tunnel 1000 will use ten units of bandwidth from the subpool.

Tunnel 2000 will use five units of bandwidth from the global pool. The “bandwidth protection desired” bit and the “node protection desired bit” have been set by specifying **bw-prot** and **node-prot**, respectively, in the **tunnel mpls traffic-eng fast-reroute** command.

```
Router(config)# interface tunnel 1000
Router(config-if)# tunnel mpls traffic-eng fast-reroute
Router(config-if)# tunnel mpls traffic-eng bandwidth sub-pool 10
Router(config)# interface tunnel 2000
Router(config-if)# tunnel mpls traffic-eng fast-reroute bw-protect node-protect
Router(config-if)# tunnel mpls traffic-eng bandwidth 5
```

Example Creating a Backup Tunnel to the Next Hop

On router R2 in the figure above, create an NHOP backup tunnel to R3. This backup tunnel should avoid using the link 10.1.1.2.

```
Router(config)# ip explicit-path name avoid-protected-link
Router(cfg-ip-expl-path)# exclude-address 10.1.1.2

Explicit Path name avoid-protected-link:
____1: exclude-address 10.1.1.2
Router(cfg-ip_expl-path)# exit

Router(config)# interface tunnel 1

Router(config-if)# ip unnumbered loopback 0

Router(config-if)# tunnel destination 10.3.3.3
Router(config-if)# tunnel mode mpls traffic-eng

Router(config-if)# tunnel mpls traffic-eng path-option 1 explicit avoid-protected-link
```

Example Creating an NNHOP Backup Tunnel

On router R2 in the figure above, create an NNHOP backup tunnel to R4. This backup tunnel should avoid R3.

```
Router(config)# ip explicit-path name avoid-protected-node

Router(cfg-ip-expl-path)# exclude-address 10.3.3.3

Explicit Path name avoid-protected-node:
____1: exclude-address 10.3.3.3
Router(cfg-ip_expl-path)# end

Router(config)# interface tunnel2

Router(config-if)# ip unnumbered loopback0

Router(config-if)# tunnel destination 10.4.4.4

Router(config-if)# tunnel mode mpls traffic-eng0

Router(config-if)# tunnel mpls traffic-eng path-option 1 explicit avoid-protected-node
```

Example Assigning Backup Tunnels to a Protected Interface

On router R2 in the figure above, both backup tunnels are associated with interface Gigabit Ethernet 0/1/0:

```
Router(config)# interface Gi0/1/0

Router(config-if)# mpls traffic-eng backup-path tunnel 1

Router(config-if)# mpls traffic-eng backup-path tunnel 2
```

Example Enabling BFD on the Protected Interface

In the figure above, BFD is enabled on interface Gigabit Ethernet 2/1/1:

```
Router(config)# interface Gi2/1/1

Router(config-if)# ip rsvp signalling hello bfd

Router(config-if)# bfd interval 100 min_rx 100 multiplier 4
```

Example Associating Backup Bandwidth and Pool Type with Backup Tunnels

In the figure above, backup tunnel 1 is to be used only by LSPs that take their bandwidth from the global pool. It does not provide bandwidth protection. Backup tunnel 2 is to be used only by LSPs that take their bandwidth from the subpool. Backup tunnel 2 provides bandwidth protection for up to 1000 units.

```
Router(config)# interface tunnel 1

Router(config-if)# tunnel mpls traffic-eng backup-bw global-pool Unlimited

Router(config)# interface tunnel 2

Router(config-if)# tunnel mpls traffic-eng backup-bw sub-pool 1000
```

Example Configuring Backup Bandwidth Protection

Note This global configuration is required only to change the backup protection preemption algorithm from minimize the number of LSPs that are demoted to minimize the amount of bandwidth that is wasted.

```
Router(config-if)# tunnel mpls traffic-eng fast-reroute bw-protect
Router(config)# mpls traffic-eng fast-reroute backup-prot-preemption optimize-bw
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
MPLS traffic engineering commands	*Cisco IOS Multiprotocol Label Switching Command Reference*
RSVP commands	*Cisco IOS Quality of Service Solutions Command Reference*
IS-IS	• *Cisco IOS IP Routing Protocols Command Reference* • Configuring a Basic IS-IS Network

Related Topic	Document Title
OSPF	- *Cisco IOS IP Routing Protocols Command Reference* - Configuring OSPF
ISSU	Cisco IOS XE In Service Software Upgrade Support
NSF/SSO	- Cisco Nonstop Forwarding - Stateful Switchover

Standards

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

MIBs

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

RFCs

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

Technical Assistance

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

Feature Information for MPLS Traffic Engineering BFD-triggered Fast Reroute

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

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

Table 3: Feature Information for MPLS Traffic Engineering: BFD-triggered Fast Reroute

Feature Name	Releases	Feature Information
MPLS Traffic Engineering: BFD-triggered Fast Reroute	12.2(33)SRC 15.3(1)S 15.1(1)SY	The MPLS Traffic Engineering: BFD-triggered Fast Reroute feature allows you to obtain link and node protection by using the Bidirectional Forwarding Detection (BFD) protocol to provide fast forwarding path failure detection times for all media types, encapsulations, topologies, and routing protocols. In addition to fast forwarding path failure detection, BFD provides a consistent failure detection method for network administrators. The following commands were introduced or modified by this feature: **clear ip rsvp hello bfd**, **ip rsvp signalling hello bfd** (configuration), **ip rsvp signalling hello bfd** (interface), **show ip rsvp hello**, **show ip rsvp hello bfd nbr**, **show ip rsvp hello bfd nbr detail**, **show ip rsvp hello bfd nbr summary**, and **show ip rsvp interface detail**.

CHAPTER 3

MPLS Traffic Engineering Nonstop Routing Support

Note This technology is not applicable for the Cisco ASR 900 RSP3 Module.

The MPLS Traffic Engineering Nonstop Routing Support feature assists the Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) routing devices to recover from an interruption in service. This feature also defines the checkpoint and recovery scheme for the devices.

Prerequisites for MPLS Traffic Engineering Nonstop Routing Support

Your network must support the following Cisco features before you enable Multiprotocol Label Switching (MPLS) Traffic Engineering (TE):

- MPLS
- Cisco Express Forwarding
- Intermediate System-to-Intermediate System (IS-IS) or Open Shortest Path First (OSPF)

Before enabling MPLS TE Nonstop Routing (NSR), a full-mode check needs to be done by the system to verify if the **mpls traffic-eng nsr** command is permitted or is restricted due to conflicts or user privileges.

Restrictions for MPLS Traffic Engineering Nonstop Routing Support

Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Nonstop Routing (NSR) and Resource Reservation Protocol (RSVP) Graceful Restart (GR) are both mutually exclusive recovery mechanisms. Hence, MPLS TE NSR cannot be enabled when RSVP GR is enabled.

Information About MPLS Traffic Engineering Nonstop Routing Support

MPLS Traffic Engineering Nonstop Routing Support Overview

Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Nonstop Routing (NSR) enables routing devices to recover from an interruption in service. The MPLS TE NSR functionality defines a checkpoint for the control plane of the routing devices. Resource Reservation Protocol (RSVP) Graceful Restart (GR) is another method for recovering and restarting interrupted services.

To avoid conflict and guarantee deterministic behavior, only one of the above mentioned recovery methods can be configured at a given time.

The MPLS TE NSR feature differs from the RSVP GR feature in the following ways:

- MPLS TE NSR devices are fully independent and do not rely on neighbor nodes for a stateful switchover (SSO) recovery.
- MPLS TE NSR supports the SSO recovery of Fast Reroute (FRR) active tunnels.
- MPLS TE NSR has an active standby mode. This helps with most of the recovery states being created before the SSO recovery actually happens, ensuring a faster recovery after SSO.
- MPLS TE NSR **show** commands display recovery information in standby mode as well.
- Label switched paths (LSPs) which are not fully signaled, do not resume signaling after an interruption and will go down on SSO.

How to Configure MPLS Traffic Engineering Nonstop Routing Support

Configuring MPLS Traffic Engineering Nonstop Routing Support

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **ip cef**
4. **mpls traffic-eng nsr**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ip cef** **Example:** `Device(config)# ip cef`	Enables standard Cisco Express Forwarding operations.
Step 4	**mpls traffic-eng nsr** **Example:** `Device(config)# mpls traffic-eng nsr`	Enables the MPLS Traffic Engineering (TE) Non-Stop Routing (NSR) functionality on a device. **Note** Enabling the MPLS TE NSR functionality automatically enables the Resource Reservation Protocol (RSVP) NSR functionality as well.

Step 5	**end**
	Example:
	`Device(config)# end`
		Exits global configuration mode and returns to privileged EXEC mode.

Verifying MPLS Traffic Engineering Nonstop Routing Support

SUMMARY STEPS

1. **enable**
2. **show mpls traffic-eng nsr**
3. **show mpls traffic-eng nsr counters**
4. **show mpls traffic-eng nsr database**
5. **show mpls traffic-eng nsr oos**
6. **show mpls traffic-eng nsr summary**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. - Enter your password if prompted.

Step 2	**show mpls traffic-eng nsr**
	Example:
	`Device# show mpls traffic-eng nsr`
	`  counters  TE NSR counters`
	`  database  TE NSR check pointed data`
	`  oos       TE NSR out of sync database`
	`  summary   TE NSR summary`
	`  |         Output modifiers`
	`  <cr>`
		Displays options to obtain Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Nonstop Routing (NSR) configuration information such as the database status, counter numbers, devices which are out of sync, and the summary of all the devices.

Step 3	**show mpls traffic-eng nsr counters** **Example:** `Device# show mpls traffic-eng nsr counters`	Displays information about the data structures or states that are successfully created or removed, along with errors counts.
Step 4	**show mpls traffic-eng nsr database** **Example:** `Device# show mpls traffic-eng nsr database`	Displays information pertaining to the write and read databases supporting MPLS TE NSR. The write and read databases store the data that is used for recovering TE state on a standby device after stateful switchover (SSO).
Step 5	**show mpls traffic-eng nsr oos** **Example:** `Device# show mpls traffic-eng nsr oos`	Displays information pertaining to the out of sync databases supporting MPLS TE NSR. The out of sync databases indicate the devices whose states are not in sync with each other.
Step 6	**show mpls traffic-eng nsr summary** **Example:** `Device# show mpls traffic-eng nsr summary`	Displays a summary of MPLS TE NSR information such as the current TE NSR state (standby-hot / recovering / staling / active), recovery time, and the recovery result (success / failure).

Step 7	**end**
	Example:
	`Device(config)# end`
		Exits privileged EXEC mode.

Configuration Examples for MPLS Traffic Engineering Nonstop Routing Support

Example: Configuring MPLS Traffic Engineering Nonstop Routing Support

The following example shows how to configure Multiprotocol (MPLS) Traffic Engineering (TE) Nonstop Routing (NSR) support on a device:

```
enable
 configure terminal
  ip cef
  mpls traffic-eng nsr
  end
```

Example: Verifying MPLS Traffic Engineering Nonstop Routing Support

Displaying MPLS Traffic Engineering Nonstop Routing Support Verification Options

The following example shows how to display the options that help you verify Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Nonstop Routing (NSR) information:

```
enable
 show mpls traffic-eng nsr ?
  counters  TE NSR counters
  database  TE NSR check pointed data
  oos       TE NSR out of sync database
  summary   TE NSR summary
  |         Output modifiers
  <cr>
```

Verifying MPLS Traffic Engineering Nonstop Routing Support Counters

The following example shows how to verify information about the data structures or states that are successfully created or removed, along with errors counts:

```
enable
 show mpls traffic-eng nsr counters

State: Active

Bulk sync
  Last bulk sync was successful (entries sent: 24)
  initiated: 1

Send timer
  started: 7

Checkpoint Messages (Items) Sent
```

```
  Succeeded:      13  (101)
    Acks accepted:13  (101)
    Acks ignored:     (0)
    Nacks:        0  (0)
  Failed:         0  (0)
  Buffer alloc:   13
  Buffer freed:   13

ISSU:
  Checkpoint Messages Transformed:
    On Send:
      Succeeded:         13
      Failed:            0
      Transformations:   0
    On Recv:
      Succeeded:         0
      Failed:            0
      Transformations:   0

  Negotiation:
    Started:              1
    Finished:             1
    Failed to Start:      0
    Messages:
      Sent:
        Send succeeded:   5
        Send failed:      0
        Buffer allocated:        5
        Buffer freed:            0
        Buffer alloc failed:     0
      Received:
        Succeeded:        7
        Failed:           0
        Buffer freed:     7
  Init:
    Succeeded:            1
    Failed:               0

  Session Registration:
    Succeeded:            0
    Failed:               0

  Session Unregistration:
    Succeeded:            0
    Failed:               0

Errors:
  None
```

Verifying MPLS Traffic Engineering Nonstop Routing Support Databases

The following example shows how to verify information pertaining to the write and read databases supporting MPLS TE NSR. The write and read databases store the data that is used for recovering TE state on a standby device after Stateful Switchover (SSO):

```
Device# show mpls traffic-eng nsr database if-autotun
IF_AUTOTUN WRITE DB

  Header:
    State: Checkpointed     Action: Add
    Seq #: 14               Flags: 0x0
  Data:
    te_nsr_seq_num: 28
    Tunnel ID: 100 (if_handle: 25), prot_if_handle: 3
    template_unit: n/a, dest: 10.2.0.1, flags=0x0

IF_AUTOTUN READ DB

Device# show mpls traffic-eng nsr database lsp-ac ?
  |  Output modifiers
  <cr>

Device# show mpls traffic-eng nsr database lsp-ac
 LM Tunnel WRITE DB:

Tun ID: 1   LSP ID: 11   (P2MP)
  SubGrp ID:   1
  SubGrp Orig: 10.1.0.1
  Dest:   10.2.0.1
  Sender: 10.1.0.1     Ext. Tun ID: 10.1.0.1
  Header:
    State: Checkpointed     Action: Add
    Seq #: 7                Flags: 0x0
    TE NSR Seq #: 14

LM Tunnel READ DB:

Device# show mpls traffic-eng nsr database internal

Write DB:
                    Checkpointed
   Entry Type       or Ack-Pending  Send-Pending
     PCALC Node                  0             0
     PCALC Link                  0             0
     PCALC Auto-Mes              0             0
     PCALC SRLG                  0             0
     lm_tunnel_t                 0             0
     NSR LSP FRR                 0             0
     nsr_if_autotun              0             0
     nsr_tspvif_set              0             0
     nsr_slsp_head               0             0

Read DB:
   Entry Type        Checkpointed
     PCALC Node                 5
     PCALC Link                12
     PCALC Auto-Mesh            0
     PCALC SRLG                 0
     lm_tunnel_t                5
     NSR LSP FRR                0
     nsr_if_autotun             0
     nsr_tspvif_setup           3
     nsr_slsp_head              5

TE NSR Sequence Bulk Sync List:
Entries: 0; next avail seq num: 132

TE NSR Sequence State Creation List:
Entries: 30; next expected seq num: 132
   Seq Num: 7  EntryPtr: 0x5A03B208
     Type: PCALC Node  Action: Add  Bundle Seq #: 1
   Seq Num: 8  EntryPtr: 0x5A0B8B38
     Type: PCALC Link  Action: Add  Bundle Seq #: 2
   Seq Num: 9  EntryPtr: 0x5A0B8DA0
```

```
     Type: PCALC Link  Action: Add  Bundle Seq #: 2
   Seq Num: 10  EntryPtr: 0x59FF1BB0
     Type: PCALC Node  Action: Add  Bundle Seq #: 1
   Seq Num: 11  EntryPtr: 0x5A0B9008
     Type: PCALC Link  Action: Add  Bundle Seq #: 2
   Seq Num: 32  EntryPtr: 0x586F2A50
     Type: PCALC Node  Action: Add  Bundle Seq #: 4
   Seq Num: 33  EntryPtr: 0x5949FC58
     Type: PCALC Link  Action: Add  Bundle Seq #: 5
   Seq Num: 34  EntryPtr: 0x5949FEC0
     Type: PCALC Link  Action: Add  Bundle Seq #: 5
   Seq Num: 60  EntryPtr: 0x5725BC30
     Type: lm_tunnel_t  Action: Add  Bundle Seq #: 12
   Seq Num: 61  EntryPtr: 0x5725BE00
     Type: nsr_tspvif_setup  Action: Add  Bundle Seq #: 12
   Seq Num: 62  EntryPtr: 0x59FC9E80
     Type: nsr_slsp_head  Action: Add  Bundle Seq #: 12
   Seq Num: 79  EntryPtr: 0x59296190
     Type: lm_tunnel_t  Action: Add  Bundle Seq #: 16
   Seq Num: 80  EntryPtr: 0x59296360
     Type: nsr_tspvif_setup  Action: Add  Bundle Seq #: 16
   Seq Num: 81  EntryPtr: 0x571EB7F8
     Type: nsr_slsp_head  Action: Add  Bundle Seq #: 16
   Seq Num: 98  EntryPtr: 0x5A04B770
     Type: lm_tunnel_t  Action: Add  Bundle Seq #: 20
   Seq Num: 99  EntryPtr: 0x59296108
     Type: nsr_tspvif_setup  Action: Add  Bundle Seq #: 20
   Seq Num: 100  EntryPtr: 0x57258670
     Type: nsr_slsp_head  Action: Add  Bundle Seq #: 20
   Seq Num: 101  EntryPtr: 0x5A060348
     Type: lm_tunnel_t  Action: Add  Bundle Seq #: 20
   Seq Num: 102  EntryPtr: 0x5A03B2B0
     Type: nsr_slsp_head  Action: Add  Bundle Seq #: 20
   Seq Num: 103  EntryPtr: 0x5B1F12B0
     Type: lm_tunnel_t  Action: Add  Bundle Seq #: 20
   Seq Num: 104  EntryPtr: 0x5A03B400
     Type: nsr_slsp_head  Action: Add  Bundle Seq #: 20
   Seq Num: 121  EntryPtr: 0x57258358
     Type: PCALC Node  Action: Add  Bundle Seq #: 21
   Seq Num: 122  EntryPtr: 0x59FAF080
     Type: PCALC Link  Action: Add  Bundle Seq #: 22
   Seq Num: 123  EntryPtr: 0x59502AC0
     Type: PCALC Link  Action: Add  Bundle Seq #: 23
   Seq Num: 124  EntryPtr: 0x594AE918
     Type: PCALC Link  Action: Add  Bundle Seq #: 21
   Seq Num: 125  EntryPtr: 0x59502120
     Type: PCALC Link  Action: Add  Bundle Seq #: 23
   Seq Num: 126  EntryPtr: 0x59FAFA20
     Type: PCALC Link  Action: Add  Bundle Seq #: 22
   Seq Num: 129  EntryPtr: 0x59FC9CC0
     Type: PCALC Node  Action: Add  Bundle Seq #: 24
   Seq Num: 130  EntryPtr: 0x5A060518
     Type: PCALC Link  Action: Add  Bundle Seq #: 24
   Seq Num: 131  EntryPtr: 0x59FAFC88
     Type: PCALC Link  Action: Add  Bundle Seq #: 24

Device# show mpls traffic-eng nsr database lsp-frr
LSP-FRR WRITE DB

Tun ID: 1   LSP ID: 10   (P2MP)
  SubGrp ID:   1
  SubGrp Orig: 10.1.0.1
  Dest:   10.2.0.1
  Sender: 10.1.0.1     Ext. Tun ID: 10.1.0.1
  Header:
    State: Checkpointed     Action: Add
    Seq #: 45               Flags: 0x0
  Data:
    te_nsr_seq_num: 164
    LSP Protected if_num: 3 (Ethernet0/0)
    LSP Next-Hop Info: rrr_id 10.2.0.1, address 10.2.0.1, label 17
    LSP Next-Next-Hop Info: rrr_id 0.0.0.0, address 0.0.0.0, label 16777216
    LSP Hold Priority: 7
    LSP bw_type: any pool
    LSP desired_bit_type: 0x0n    LSP Backup ERO address 10.1.2.2
    LSP advertise_bw: NO

LSP-FRR READ DB

Device# show mpls traffic-eng nsr database lsp-frr filter destination ?
  Hostname or A.B.C.D  IP addr or name of destination (tunnel tail)

Device# show mpls traffic-eng nsr database lsp-frr filter lsp-id ?
  <0-65535>  LSP ID

Device# show mpls traffic-eng nsr database lsp-frr filter source ?
  Hostname or A.B.C.D  IP addr or name of sender (tunnel head)

Device# show mpls traffic-eng nsr database lsp-frr filter tunnel-id ?
  <0-65535>  tunnel ID

Device# show mpls traffic-eng nsr database lsp-head
SLSP_HEAD WRITE DB

  Tun ID: 0  (P2P), lsp_id: 7
  Header:
    State: Checkpointed     Action: Add
    Seq #: 6                Flags: 0x0
  Data:
    te_nsr_seq_num: 18
    bandwidth: 5, thead_flags: 0x1, popt: 1
    feature flags: none
    output_if_num: 11, output_nhop: 10.1.3.2
    backup_output_if_num: 0
    output_tag: 19
    backup_output_tag: 16777218
    RRR path setup info
      Destination: 10.3.0.1, Id: 10.3.0.1 Router Node (ospf) flag:0x0
      IGP: ospf, IGP area: 0, Number of hops: 3, metric: 128
      Hop 0: 10.1.3.2, Id: 10.2.0.1 Router Node (ospf), flag:0x0
      Hop 1: 10.2.3.3, Id: 10.3.0.1 Router Node (ospf), flag:0x0
      Hop 2: 10.3.0.1, Id: 10.3.0.1 Router Node (ospf), flag:0x0

SLSP_HEAD READ DB

Device# show mpls traffic-eng nsr database lsp-head filter destination ?
  Hostname or A.B.C.D  IP addr or name of destination (tunnel tail)

Device# show mpls traffic-eng nsr database lsp-head filter lsp-id ?
  <0-65535>  LSP ID

Device# show mpls traffic-eng nsr database lsp-head filter source ?
  Hostname or A.B.C.D  IP addr or name of sender (tunnel head)

Device# show mpls traffic-eng nsr database lsp-head filter tunnel-id ?
  <0-65535>  tunnel ID

Device# show mpls traffic-eng nsr database pcalc auto-mesh
```

```
  PCALC Auto-Mesh WRITE DB:

  PCALC Auto-Mesh READ DB:

Device# show mpls traffic-eng nsr database pcalc nbr
 PCALC Link WRITE DB:
  Header:
    State: Checkpointed     Action: Add
    Seq #: 4                Flags: 0x0
    TE NSR Seq #: 26
    IGP Id:10.1.2.2     Area:0   Nbr IGP Id:10.1.2.2
    IP:10.1.2.1         Nbr IP:0.0.0.0  Framgment ID:1
    Intf ID    Local:0     Remote:0


 PCALC Link READ DB:

Device# show mpls traffic-eng nsr database pcalc node
 PCALC Node WRITE DB:
  Header:
    State: Checkpointed     Action: Add
    Seq #: 4                Flags: 0x0
    TE NSR Seq #: 25
    Router Id 10.1.0.1
    node_id 1
    num_links 2
    tlvs_len 0
    flags 0x6
    rid_frag_id 0
    bcid_mismatch 0
    incarnation  0

Device# show mpls traffic-eng nsr database pcalc srlg
 PCALC SRLGs WRITE DB:

 PCALC SRLGs READ DB:

Device# show mpls traffic-eng nsr database summary
MPLS-TE Non-Stop-Routing is ENABLED

Write DB Coalescing: INACTIVE
Write DB:
  Send-Pending:     0
  Ack-Pending :     0
  Checkpointed:    35
  Total       :    35

Read DB:
  Total       :     0

Device# show mpls traffic-eng nsr database tun-setup
TSPVIF_SETUP WRITE DB

  Tun ID: 0, lsp_id: 7
  Header:
    State: Checkpointed     Action: Add
    Seq #: 6                Flags: 0x0
  Data:
    te_nsr_seq_num: 17
    Setup Evt: allocating current tspsetup, chkpt_flags: 0x0

TSPVIF_SETUP READ DB
```

Verifying MPLS Traffic Engineering Nonstop Routing Support Out-of-Sync Databases

The following example shows how to verify information pertaining to the out-of-sync databases supporting MPLS TE NSR. The out-of-sync databases indicate the **active and standby RSP** whose states are not in sync with each other:

```
enable
 show mpls traffic-eng nsr oos
  Tunnel: 4000
  Time created: 02/20/13-12:03:13
  Time synced: 02/20/13-12:03:14
  Key:
    Source:                    10.1.0.1
    Destination:               10.2.0.1
    ID:                        4000
    Ext Tun ID:                10.1.0.1
    Instance:                  4
    Slsp p2mp ID:              0
    Slsp p2mp subgroup ID:     0
    Slsp p2mp subgroup origin: 0

  RSVP States:
    Signal:       Unknown
    Fast-Reroute: Disabled
    Delete State: True

  TE States:
    Signal:       Unknown
    Fast-Reroute: Disabled
    Delete State: True

  Update History:
    Total number of updates: 2

      Update Time: 02/20/13-12:03:13
        Client Updating: RSVP
        Update State:
          Signal:       Unknown
          Fast-Reroute: Unknown
          Delete State: True

      Update Time: 02/20/13-12:03:14
        Client Updating: TE
        Update State:
          Signal:       Unknown
          Fast-Reroute: Unknown
          Delete State: True
```

Verifying MPLS Traffic Engineering Nonstop Routing Support Information Summary

The following example shows how to view a summary of MPLS TE NSR information such as the current TE NSR state (standby-hot / recovering / staling / active), recovery time, and the recovery result (success / failure):

```
enable
 show mpls traffic-eng nsr summary
  State:
```

```
Graceful-Restart: Disabled
HA state: Active
Checkpointing: Allowed
Messages:
 Send timer: not running (Interval: 1000 msec)
 Items sent per Interval: 200
 CF buffer size used: 3968
```

Additional References for MPLS Traffic Engineering Nonstop Routing Support

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Multiprotocol Label Switching High Availability Configuration Guide	Cisco IOS XE Multiprotocol Label Switching High Availability Configuration Guide
MPLS TE commands	Cisco IOS Multiprotocol Label Switching Command Reference

Standards and RFCs

Standard/RFC	Title
RFC 2205	*Resource Reservation Protocol (RSVP)*

Technical Assistance

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

Feature Information for MPLS Traffic Engineering Nonstop Routing Support

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

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

Table 4: Feature Information for MPLS Traffic Engineering Nonstop Routing Support

Feature Name	Releases	Feature Information
MPLS Traffic Engineering Nonstop Routing Support	Cisco IOS XE Release 3.10S, 3.13S	The MPLS Traffic Engineering Non-Stop Routing Support feature assists the Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) routing devices to recover from an interruption in service. The MPLS TE Nonstop Routing (NSR) support functionality also defines the checkpoint and recovery scheme for the devices. From Cisco IOS XE 3.13S, support was provided for ASR 903. The following commands were introduced: **mpls traffic-eng nsr** and **show mpls traffic-eng nsr**.

CHAPTER 4

MPLS Traffic Engineering over Bridge Domain Interfaces

The MPLS Traffic Engineering(TE) over Bridge Domain Interfaces(BDI) feature enables MPLS traffic engineering over Bridge Domain Interfaces.

Prerequisites for Configuring MPLS TE over BDI

You must have:

- Enabled MPLS TE on all relevant routers and interfaces
- Configured MPLS TE tunnels

Your network must support the following Cisco IOS features:

- IP Cisco Express Forwarding
- Multiprotocol Label Switching (MPLS)

Your network must support at least one of the following protocols:

- Intermediate SystemtoIntermediate System (ISIS)
- Open Shortest Path First (OSPF)

Restrictions for MPLS TE over BDI

- MPLS TE - Verbatim Path Support
- Explicit Path Node exclusion
- P2MP TE Tunnels
- Auto-tunnel one-hops and backups

- Auto bandwidth
- Inter area or AS TE
- Auto route destinations
- FRR link ornode protection

Information About MPLS Traffic Engineering over BDI

Features of MPLS Traffic Engineering over BDI

The MPLS Traffic Engineering over BDI feature enables MPLS TE tunnels over BDI.

Supported Features

Your network must support the following:

- MPLS TE tunnels
- Policy Routing onto MPLS TE Tunnels
- MPLS TE - Forwarding Adjacency
- MPLS TE – RSVP Hello State Timer
- MPLS TE - LSP Attributes
- MPLS TE - IP Explicit Address Exclusion
- MPLS TE - Configurable Path Calculation Metric for Tunnels
- MPLS TE - Verbatim Path Support
- Pseudo-wire mapping onto TE tunnels.

How to Configure MPLS Traffic Engineering over BDI

This section assumes that you want to configure MPLS TE over BDI.

Configuring MPLS TE over BDI

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface bdi30**
4. **mpls traffic-eng tunnels**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters interface configuration mode.
Step 3	**interface bdi30** **Example:** `Router(config)# interface bdi30`	Specifies the bridge domain interface and enters interface configuration mode.
Step 4	**mpls traffic-eng tunnels** **Example:** `Router(config-if)# mpls traffic-eng tunnels`	Enables an MPLS TE tunnel to use an established tunnel for the bridge domain interface.

Step 5	**end**
	Example:
	`Router(config-if)# end`
		Returns to privileged EXEC mode.

Configuring the RSVP Bandwidth

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type slot* / *subslot* / *port*
4. **ip rsvp bandwidth** [*interface-kbps* [*single-flow-kbps*[**bc1** *kbps* | **sub-pool** *kbps*]*]*| **percent** *percent-bandwidth* [*single-flow-kbps*]]
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type slot* **/** *subslot* **/** *port* **Example:** `Router(config)# interface gigabitEthernet 0/0/0`	Configures the interface type and enters interface configuration mode.
Step 4	**ip rsvp bandwidth** [*interface-kbps* [*single-flow-kbps*[**bc1** *kbps* | **sub-pool** *kbps*]*]*| **percent** *percent-bandwidth* [*single-flow-kbps*]] **Example:** `Router(config-if)# ip rsvp bandwidth 7500 7500`	Enables RSVP on an interface. • The optional *interface-kbps* and *single-flow-kbps* arguments specify the amount of bandwidth that can be allocated by RSVP flows or to a single flow, respectively. Values are from 1 to 10000000. • The optional **sub-pool**and *kbps*keyword and argument specify subpool traffic and the amount of bandwidth that can be allocated by RSVP flows. Values are from 1 to 10000000. **Note** Repeat this command for each interface on which you want to enable RSVP.

Step 5	**end**
	Example:
	`Router(config-if)# end`
		(Optional) Returns to privileged EXEC mode.

Verifying That MPLS TE over BDI Is Operational

To verify that MPLS TE over BDI can function, perform the following task.

SUMMARY STEPS

1. **enable**
2. **show mpls traffic-eng tunnels brief**
3. **show mpls traffic-eng tunnels summary**
4. **show mpls traffic-eng tunnels tunnel1**

DETAILED STEPS

Step 1 **enable**

Enables privileged EXEC mode.

Step 2 **show mpls traffic-eng tunnels brief**

Use this command to monitor and verify the state of the tunnels.

Step 3 **show mpls traffic-eng tunnels summary**

Use this command to monitor and verify the state of the tunnels.

Step 4 **show mpls traffic-eng tunnels tunnel1**

Use this command to verify that tunnels are up and using BDI.

Troubleshooting Tips

This section describes how you can use the `show mpls traffic-eng tunnels tunnel5` to check for issues.

```
Router# show mpls traffic-eng tunnels tunnel5

Name: router_t5                           (Tunnel5) Destination: 3.3.3.3
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 1, type dynamic (Basis for Setup, path weight 2)

  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 5  5   Affinity: 0x0/0xFFFF
    Metric Type: TE (default)
    AutoRoute: disabled LockDown: disabled Loadshare: 0 [0] bw-based
    auto-bw: disabled
  Active Path Option Parameters:
    State: dynamic path option 1 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled


  InLabel  :  -
  OutLabel : BDI31, 21
  Next Hop : 12.0.0.2
  RSVP Signalling Info:
       Src 1.1.1.1, Dst 3.3.3.3, Tun_Id 5, Tun_Instance 1
    RSVP Path Info:
      My Address: 12.0.0.1
      Explicit Route: 12.0.0.2 14.0.0.2 14.0.0.1 3.3.3.3
      Record   Route:   NONE
      Tspec: ave rate=0 kbits, burst=1000 bytes, peak rate=0 kbits
    RSVP Resv Info:
      Record   Route:   NONE
      Fspec: ave rate=0 kbits, burst=1000 bytes, peak rate=0 kbits
  Shortest Unconstrained Path Info:
    Path Weight: 2 (TE)
    Explicit Route: 12.0.0.1 12.0.0.2 14.0.0.2 14.0.0.1
                    3.3.3.3
  History:
    Tunnel:
      Time since created: 1 minutes, 38 seconds
      Time since path change: 1 minutes, 36 seconds
      Number of LSP IDs (Tun_Instances) used: 1
    Current LSP: [ID: 1]
      Uptime: 1 minutes, 36 seconds
```

Configuration Example for MPLS Traffic Engineering over BDI

The following example enables the BDI on the router:

```
Router(config)#interface bdi30
Router(config-if)#mpls traffic-eng tunnels
```

Configuring Interface Tunnel Example

The following example configures an interface tunnel

```
interface Tunnel1
ip unnumbered Loopback0
tunnel source Loopback0
tunnel mode mpls traffic-eng
tunnel destination 4.4.4.4
tunnel mpls traffic-eng path-option 1 dynamic
```

Configuring RSVP Bandwidth Example

The following example configures RSVP bandwidth

ip rsvp bandwidth [*interface-kbps*] [*single-flow-kbps*]

```
Router(config-if)# ip rsvp bandwidth 500 500
```

CHAPTER 5

MPLS Point-to-Multipoint Traffic Engineering Support for Static Pseudowires

Table 5: Feature History

Feature Name	Release	Description
Static PW over P2MP	Cisco IOS XE Amsterdam 17.3.1	The Static Pseudowires over Point-to-Multipoint Traffic Engineering (P2MP TE) feature emulates the essential attributes of a unidirectional P2MP service. It can be used to transport layer 2 multicast services from a single source to one or more destinations. This feature is supported on the Cisco RSP2 module. This feature is supported on the Cisco RSP3 module.

The MPLS Point-to-Multipoint Traffic Engineering: Support for Static Pseudowires feature allows you to configure a point-to-multipoint pseudowire (PW) to transport Layer 2 traffic from a single source to one or more destinations. This feature provides traffic segmentation for Multiprotocol Label Switching (MPLS) Point-to-Multipoint Traffic Engineering (P2MP TE) tunnels.

The MPLS Point-to-Multipoint Traffic Engineering: Support for Static Pseudowires feature uses Layer 2 Virtual Private Network (L2VPN) static PWs to provide point-to-multipoint Layer 2 connectivity over an MPLS network to transport Layer 2 traffic. The static PW does not need Label Distribution Protocol (LDP).

Prerequisites for MPLS Point-to-Multipoint Traffic Engineering Support for Static Pseudowires

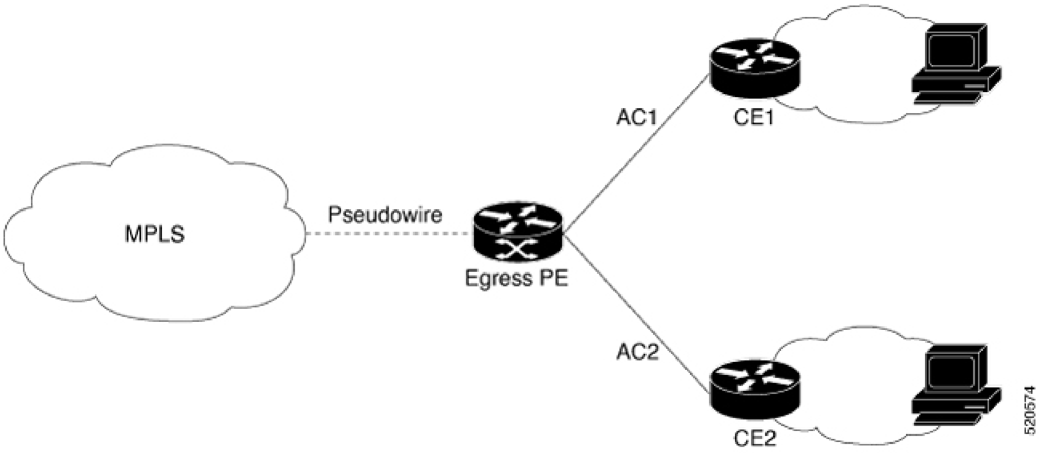
Before configuring the MPLS Point-to-Multipoint Traffic Engineering: Support for Static Pseudowires feature, ensure that the following prerequisite is met:

- If a Cisco RSP3 module acts as a P2MP TE midpoint, it should be running the Cisco IOS XE Release 17.3.1 or later releases.

Restrictions for MPLS Point-to-Multipoint Traffic Engineering Support for Static Pseudowires

- Only EVC-based Ethernet over MPLS is supported. TDM MPLS is **not** supported.
- Multiple Xconnects cannot be configured with same P2MP Tunnel as it leads to traffic drop for one of the Connects.
- If the preferred-paths under pseudowire-class of the Xconnects are swapped, Xconnect interface should be flapped to resume traffic.
- P2MP Tunnel cannot be used to forward Static PW traffic and Global IPv4 multicast traffic (MVPN profile 8) simultaneously.
- Static PW over P2MP is standardized as unidirectional. But the current configuration model does not block packet forwarding from the receiver to the source.
- Local bindings must be unique. Otherwise traffic will accidentally merge.
- Replication of egress is not supported. Only a single CE connects to a PE which is part of the Tunnel destination list of one P2MP Pseudowire.
- Effective Cisco IOS XE 17.3.1, the Static PW over Point-to-Multipoint tunnel can be scaled up to 400 tunnels and 400 Static PWs.

Figure 10: Egress Packet Replication



Note You must use the **no show ip rsvp** command to check tunnel bandwidth. If the total tunnel bandwidth exceeds beyond 750 MB (megabits per second), then the sub-LSPs go down when toggling the **traffic-eng** command with a maximum tunnel bandwidth of 749.9 MB.

For example, there are 82 P2MP tunnels and you configure 9146 kbps for each tunnel. Then the total bandwidth allocated is (9146 kbps * 82 tunnels) = 749.9 MB.

Information About MPLS Point-to-Multipoint Traffic Engineering Support for Static Pseudowires

Overview of MPLS Point-to-Multipoint Traffic Engineering Support for Static Pseudowires

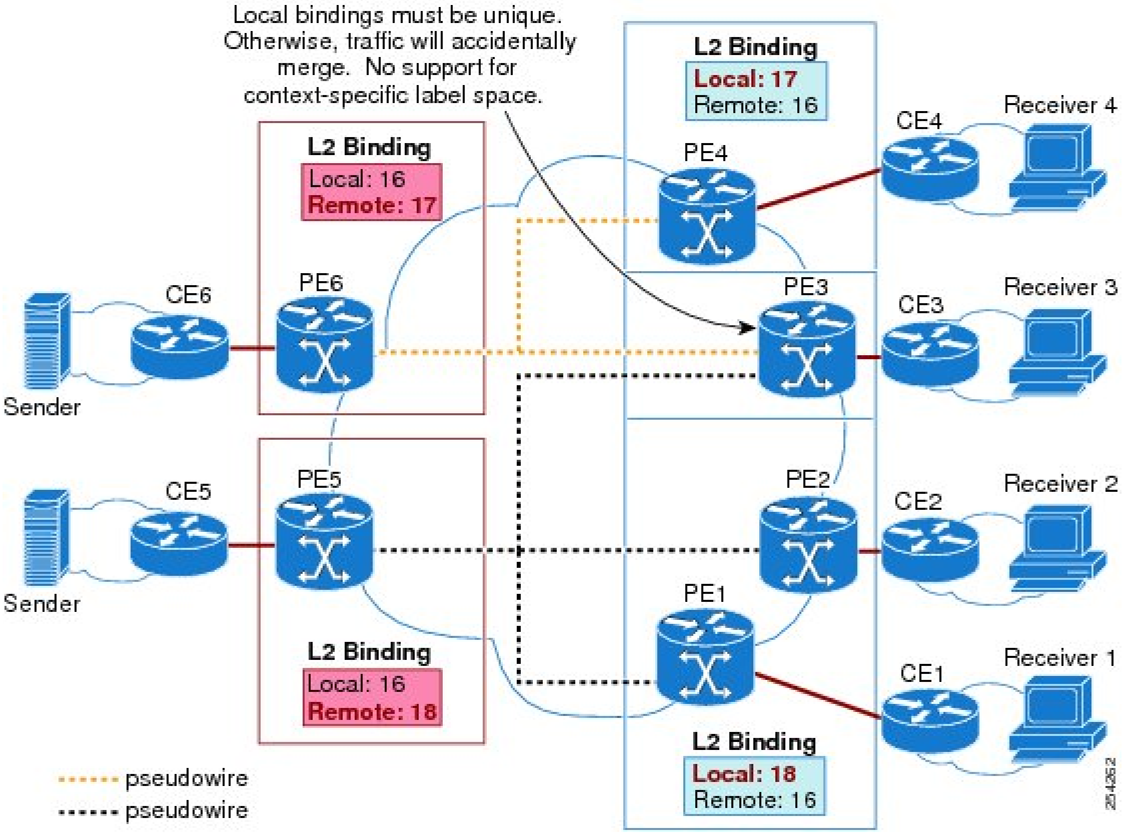
The MPLS Point-to-Multipoint Traffic Engineering: Support for Static Pseudowires feature transports Layer 2 traffic from a single source to one or more destinations. This feature has the following characteristics:

- It uses L2VPN static PWs to provide point-to-multipoint Layer 2 connectivity over an MPLS network to transport Layer 2 traffic.
- The segmentation for MPLS P2MP TE tunnels provided by this feature allows for applications such as video distribution and clock distribution (mobile backhaul).
- This feature is compatible with Cisco nonstop forwarding (NSF), stateful switchover (SSO). See NSF/SSO—MPLS TE and RSVP Graceful Restart and MPLS Point-to-Multipoint Traffic Engineering for information on configuring NSF/SSO with this feature.
- In this implementation, the PW is bidirectional, in accordance with the Framework and Requirements for Virtual Private Multicast Service .

VC Label Collisions

This feature does not support context-specific label spaces. When configuring the MPLS Point-to-Multipoint Traffic Engineering: Support for Static Pseudowires feature, ensure that local bindings are unique. Otherwise, traffic unintentionally merges. In the figure below, both PWs share router PE 3 as an endpoint. The local label on each PW is 16, which causes a collision.

Figure 11: Avoiding VC Label Collisions



Label Spoofing

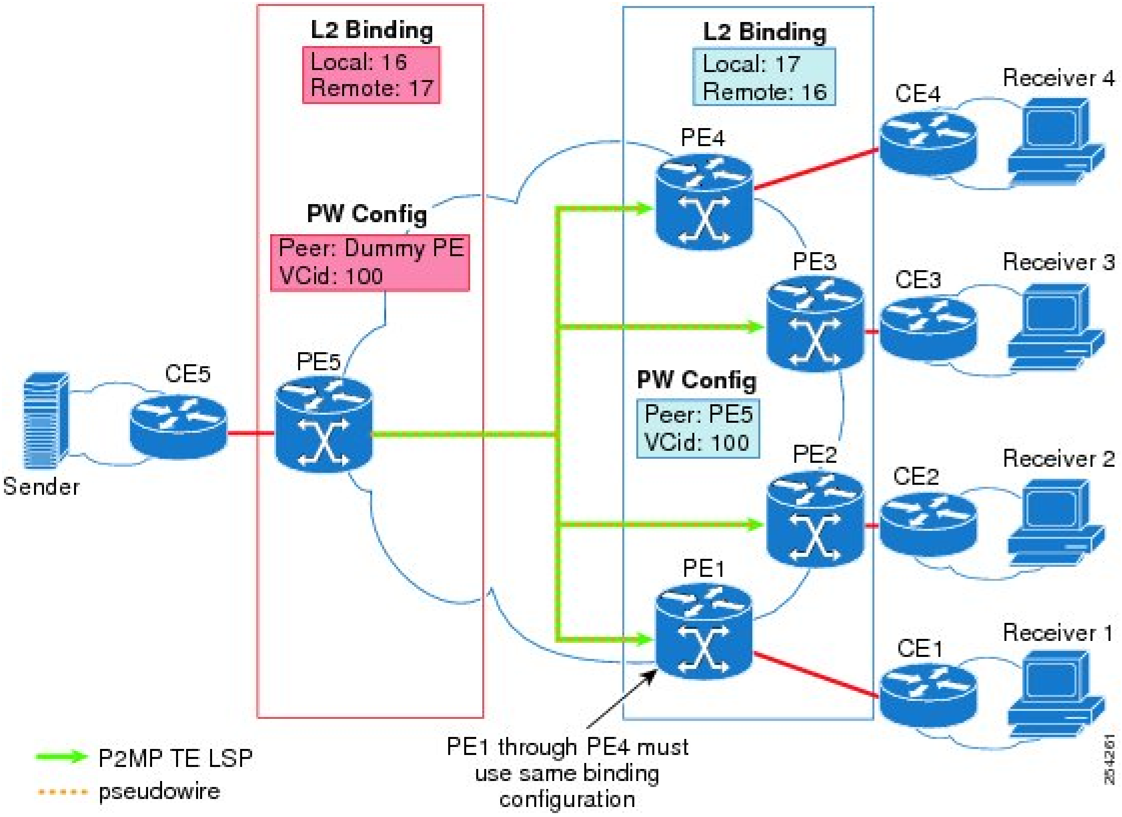
For P2MP static PWs, there is no signaling protocol to verify that the labels are configured correctly on either end. If the labels are not configured correctly, traffic might go to the wrong destinations. Because the traffic going into wrong destinations is a multicast confutation, scalability might be impacted.

The P2MP static PW does not have a context-specific label in the upstream direction and does not use a signaling protocol. Therefore, it is possible to spoof a PW label and route the traffic to the wrong destination. If a PW label is spoofed at the headend, it cannot be validated at the tailend, because the MPLS lookup at the tailend is performed on the global table. So if a spoofed label exists in the global table, traffic is routed to the wrong destination: customer equipment (CE).

The same situation can happen if the user incorrectly configures the static PW label. If the wrong PW label is configured, traffic goes to the wrong destination (CE).

The figure below shows PW label allocation with no context-specific label space.

Figure 12: PW Label Allocation with No Context-Specific Label Space



How to Configure MPLS Point-to-Multipoint Traffic Engineering Support for Static Pseudowires

Configuring the MPLS Label Range

You must specify a static range of MPLS labels using the **mpls label range** command with the **static** keyword.

SUMMARY STEPS

1. enable
2. configure terminal
3. mpls label range *minimum-value maximum-value{* **static***minimum-static-value maximum-static-value}*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	`Router> enable`	
Step 2	configure terminal **Example:** `Router# configure terminal`	Enters global configuration mode.

Step 3	mpls label range *minimum-value maximum-value{* **static***minimum-static-value maximum-static-value}*
	Example:
	`Router(config)# mpls label range 1001 1003 static 10000 25000`
		Specifies a static range of MPLS labels

Configuring the Headend Routers

Perform this task to configure the headend routers:

- MPLS Static Label range must be configured to configure the Static PW Label under Xconnect.
- Under the Pseudowire class, the P2MPTE tunnel interface should be specified as the preferred path.
- 172.10.255.255 is a fake peer IP address. It is very important that this IP address be reserved by the network domain administrator so that it is not used by any other routers in the network.
- Instead of a fake peer IP address, if peer IP address is used that is present in the routing table, then the traffic will flow through the LSP path (formed by the LDP) towards the peer. This happens only if the fallback option under pseudowire class is not disabled (default) and the preferred path is down.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **pseudowire-class** *class-name*
4. **encapsulation mpls**
5. **protocol none**
6. **preferred-path** [**interfacetunnel***tunnel-number*][**disable-fallback**]
7. **exit**
8. **interface tunnel** *number*
9. **ip unnumbered loopback** *number*
10. **tunnel mode mpls traffic-eng point-to-multipoint**
11. **tunnel destination list mpls traffic-eng** {**identifier***dest-list-id* | **name***dest-list-name*}
12. **exit**
13. **interface loopback** *number*
14. **ip address** [*ip-addressmask* [**secondary**]]
15. **exit**
16. **interface ethernet** *number*
17. **no ip address** [*ip-addressmask* [**secondary**]]
18. **no keepalive** [*period* [*retries*]]
19. **xconnect** *peer-ip-address vcid* **encapsulation mpls manual pw-class** *class-name*

20. **mpls label** *local-pseudowire-label remote-pseudowire-label*
21. **mpls control-word**
22. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**pseudowire-class** *class-name* **Example:** `Router(config)# pseudowire-class static-pw`	**S** pecifies a static AToM PW class and enters PW class configuration mode.
Step 4	**encapsulation mpls** **Example:** `Router(config-pw)# encapsulation mpls`	Specifies MPLS as the data encapsulation method for tunneling Layer 2 traffic over the PW.
Step 5	**protocol none** **Example:** `Router(config-pw)# protocol none`	**S** pecifies that no signaling will be used in L2TPv3 sessions created from the static PW.
Step 6	**preferred-path** [**interfacetunnel***tunnel-number*][**disable-fallback**] **Example:** `Router(config-pw)# preferred-path interface tunnel 1 disable-fallback`	Specifies the P2MP tunnel as the traffic path and disables the router from using the default path when the preferred path is unreachable.
Step 7	**exit** **Example:** `Router(config-pw)# exit`	ExitsPW class configuration mode and returns to global configuration mode.
Step 8	**interface tunnel** *number* **Example:** `Router(config)# interface tunnel 1`	Configures a tunnel and enters interface configuration mode.

	Command or Action	Purpose
Step 9	**ip unnumbered loopback** *number* **Example:** `Router(config-if)# ip unnumbered loopback 0`	Enables IP processing on a loopback interface without assigning an explicit IP address to the interface. • Specifying loopback 0 gives the tunnel interface an IP address that is the same as that of loopback interface 0. • This command is not effective until loopback interface 0 has been configured with an IP address.
Step 10	**tunnel mode mpls traffic-eng point-to-multipoint** **Example:** `Router(config-if)# tunnel mode mpls traffic-eng point-to-multipoint`	Enables MPLS P2MP TE on the tunnel.
Step 11	**tunnel destination list mpls traffic-eng** {**identifier***dest-list-id* | **name***dest-list-name*} **Example:** `Router(config-if)# tunnel destination list mpls traffic-eng name in-list-01`	Specifies a destination list to specify the IP addresses of point-to-multipoint destinations.
Step 12	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode and returns to global configuration mode.
Step 13	**interface loopback** *number* **Example:** `Router(config)# interface loopback 0`	Configures a loopback interface and enters interface configuration mode.
Step 14	**ip address** [*ip-addressmask* [**secondary**]] **Example:** `Router(config-if)# ip address 172.16.255.5 255.255.255.255`	Specifies a primary IP address for the loopback interface.
Step 15	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode and returns to global configuration mode.
Step 16	**interface ethernet** *number* **Example:** `Router(config)# interface ethernet 0/0/0`	Configures an Ethernet interface and enters interface configuration mode.

	Command or Action	Purpose

Step 17	**no ip address** [*ip-addressmask* [**secondary**]]
	Example:
	`Router(config-if)# no ip address`
	`service instance 100 ethernet`
		Disables IP processing on the interface.

Step 18	**no keepalive** [*period* [*retries*]] **Example:** `Router(config-if)# no keepalive`	Disables the keepalive packets on the interface. • When the interface goes down, the session continues without shutting down because the keepalive packets are disabled.

Step 19	**xconnect** *peer-ip-address vcid* **encapsulation mpls manual pw-class** *class-name*
	Example:
	`Router(config-if)# xconnect 172.16.255.255 100`
	`encapsulation mpls manual pw-class static-pw`
		Configures a static AToM PW and enters xconnect configuration mode where the static PW labels are set.

Step 20	**mpls label** *local-pseudowire-label remote-pseudowire-label* **Example:** `Router(config-if-xconn)# mpls label 16 17`	Configures the AToM static PW connection by defining local and remote circuit labels. • The label must be an unused static label within the static label range configured using the **mplslabelrange** command. • The **mplslabel**command checks the validity of the label entered and displays an error message if it is not valid. The value supplied for the *remote-pseudowire-label*argument must be the value of the peer PE's local PW label.
Step 21	**mpls control-word** **Example:** `Router(config-if-xconn)# mpls control-word`	Checks whether the MPLS control word is sent. • This command must be set for Frame Relay data-link connection identifier (DLCI) and ATM adaptation layer 5 (AAL5) attachment circuits. For other attachment circuits, the control word is included by default. • If you enable the inclusion of the control word, it must be enabled on both ends of the connection for the circuit to work properly. • Inclusion of the control word can be explicitly disabled using the **nomplscontrol-word**command.

Step 22	**end**
	Example:
	`Router(config-if-xconn)# end`
		Exits xconnect configuration mode.

Configuring the MPLS Label Range

You must specify a static range of MPLS labels using the **mpls label range** command with the **static** keyword.

SUMMARY STEPS

1. enable
2. configure terminal
3. mpls label range *minimum-value maximum-value{* **static***minimum-static-value maximum-static-value}*

DETAILED STEPS

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

Step 3	mpls label range *minimum-value maximum-value{* **static***minimum-static-value maximum-static-value}*
	Example:
	`Router(config)# mpls label range 1001 1003 static 10000 25000`
		Specifies a static range of MPLS labels

Configuring the Tailend Routers

Perform this task to configure the tailend routers:

- MPLS Static Label range should be configured to configure Static PW label under Xconnect.
- The loopback address of the headend router (source of the tree) should be configured under the Xconnect of all tailend routers. The loopback address of the headend router in this example is 1.1.1.1
- All tailend routers should be configured with same remote Virtual Circuit (VC) label of 200, the local VC label of the headend router.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **pseudowire-class** *class-name*
4. **encapsulation mpls**
5. **protocol none**
6. **exit**
7. **interface loopback** *number*
8. **ip address** [*ip-addressmask* [**secondary**]]

9. **exit**
10. **interface ethernet** *number*
11. **no ip address** [*ip-addressmask* [**secondary**]]
12. **no keepalive** [*period* [*retries*]]
13. **xconnect** *peer-ip-address vcid* **encapsulation mpls manual pw-class** *class-name*
14. **mpls label** *local-pseudowire-label remote-pseudowire-label*
15. **mpls control-word**
16. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**pseudowire-class** *class-name* **Example:** `Router(config)# pseudowire-class static-pw`	Specifies a static AToM PW class and enters PW class configuration mode.
Step 4	**encapsulation mpls** **Example:** `Router(config-pw)# encapsulation mpls`	Specifies MPLS as the data encapsulation method for tunneling Layer 2 traffic over the PW.
Step 5	**protocol none** **Example:** `Router(config-pw)# protocol none`	Specifies that no signaling will be used in L2TPv3 sessions created from the static PW.
Step 6	**exit** **Example:** `Router(config-pw)# exit`	ExitsPW class configuration mode and returns to global configuration mode.
Step 7	**interface loopback** *number* **Example:** `Router(config)# interface loopback 0`	Configures a loopback interface and enters interface configuration mode.

	Command or Action	Purpose
Step 8	**ip address** [*ip-addressmask* [**secondary**]] **Example:** `Router(config-if)# ip address 172.16.255.1 255.255.255.255`	Specifies a primary IP address for the loopback interface.
Step 9	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode and returns to global configuration mode.
Step 10	**interface ethernet** *number* **Example:** `Router(config)# interface ethernet 0/0`	Configures an Ethernet interface and enters interface configuration mode.
Step 11	**no ip address** [*ip-addressmask* [**secondary**]] **Example:** `Router(config-if)# no ip address`	Disables IP processing on the interface.
Step 12	**no keepalive** [*period* [*retries*]] **Example:** `Router(config-if)# no keepalive`	Disables the keepalive packets on the interface. • When the interface goes down, the session continues without shutting down because the keepalive packets are disabled.
Step 13	**xconnect** *peer-ip-address vcid* **encapsulation mpls manual pw-class** *class-name* **Example:** `Router(config-if)# xconnect 172.16.255.5 100 encapsulation mpls manual pw-class static-pw`	Configures a static AToM PW and enters xconnect configuration mode where the static PW labels are set.
Step 14	**mpls label** *local-pseudowire-label remote-pseudowire-label* **Example:** `Router(config-if-xconn)# mpls label 17 16`	Configures the AToM static PW connection by defining local and remote circuit labels. • The label must be an unused static label within the static label range configured using the **mplslabelrange** command. • The**mplslabel**command checks the validity of the label entered and displays an error message if it is not valid. The value supplied for the *remote-pseudowire-label*argument must be the value of the peer PE’s local PW label.
Step 15	**mpls control-word** **Example:**	Checks whether the MPLS control word is sent.

	Command or Action	Purpose
	`Router(config-if-xconn)# mpls control-word`	• This command must be set for Frame Relay data-link connection identifier (DLCI) and ATM adaptation layer 5 (AAL5) attachment circuits. For other attachment circuits, the control word is included by default. • If you enable inclusion of the control word, it must be enabled on both ends of the connection for the circuit to work properly. • Inclusion of the control word can be explicitly disabled using the **nomplscontrol-word** command.

Step 16	**end**
	Example:
	`Router(config-if-xconn)# end`
		Exits xconnect configuration mode.

Verifying the Static PW Configuration

To verify the L2VPN static PW configuration, use the **showrunning-config** EXEC command. To verify that the L2VPN static PW was provisioned correctly, use the **showmplsl2transportvcdetail**and **pingmplspseudowire**EXEC commands as described in the following steps.

SUMMARY STEPS

1. **show mpls l2transport vc detail**

DETAILED STEPS

show mpls l2transport vc detail

For nonstatic PW configurations, this command lists the type of protocol used to send the MPLS labels (such as LDP). For static PW configuration, the value of the signaling protocol field should be Manual.

The following is sample output from the **showmplsl2transportvcdetail**command:

Example:

```
PE21_RSP2#sh mpls l2transport vc 3750 detail
Local interface: Gi0/2/5 up, line protocol up, Eth VLAN 3750 up
  Destination address: 172.10.255.255, VC ID: 3750, VC status: up
    Output interface: Tu3750, imposed label stack {}
    Preferred path: Tunnel3750,  active
    Default path: ready
    No adjacency
  Create time: 3d20h, last status change time: 3d20h
    Last label FSM state change time: 3d20h
  Signaling protocol: Manual
    Status TLV support (local/remote)   : disabled/N/A
      LDP route watch                   : enabled
      Label/status state machine        : established, LruRru
      Last local dataplane   status rcvd: No fault
```

```
        Last BFD dataplane     status rcvd: Not sent
        Last BFD peer monitor  status rcvd: No fault
        Last local AC  circuit status rcvd: No fault
        Last local AC  circuit status sent: No fault
        Last local PW i/f circ status rcvd: No fault
        Last local LDP TLV     status sent: No status
        Last remote LDP TLV    status rcvd: Not sent
        Last remote LDP ADJ    status rcvd: No fault
      MPLS VC labels: local 10750, remote 11750
      Group ID: local 21, remote 21
      MTU: local 1500, remote 1500
    Sequencing: receive disabled, send disabled
    Control Word: On (configured: autosense)
    SSO Descriptor: 172.10.255.255/3750, local label: 10750
    Dataplane:
      SSM segment/switch IDs: 1008461/4594 (used), PWID: 112
  VC statistics:
      transit packet totals: receive 0, send 105053403
      transit byte totals:   receive 0, send 53787342336
      transit packet drops:  receive 0, seq error 0, send 0
```

Configuration Examples for MPLS Point-to-Multipoint Traffic Engineering Support for Static Pseudowires

Example Configuring the Headend Router (PE5)

In the following sample configuration of the headend router, note the following:

- The **preferred-pathinterfacetunnel1**command specifies the P2MP tunnel as the preferred path.
- The **tunnelmodemplstraffic-engpoint-to-multipoint** command enables the P2MP tunnel.
- The **mplslabel**command defines the static binding.
- The **xconnect**command creates a dummy peer.

```
Router(config)# pseudowire-class STATIC-PW
Router(config-pw-class)# encapsulation mpls
Router(config-pw-class)# protocol none
Router(config-pw-class)# preferred-path interface Tunnel1

!
Router(config)# interface Tunnel1
Router(config-if)# description PE5->PE1,PE2,PE3,PE4-EXCIT
Router(config-if)# ip unnumbered loopback 0
Router(config-if)# tunnel mode mpls traffic-eng point-to-multipoint
Router(config-if)# tunnel destination list mpls traffic-eng name P2MP-EXCIT-DST-LIST
Router(config-if)# tunnel mpls traffic-eng priority 7 7
Router(config-if)# tunnel mpls traffic-eng bandwidth 10000
!
Router(config)# interface loopback 0
Router(config-if)# ip address 172.16.255.5 255.255.255.255
!
Router(config)# interface ethernet 0/0
Router(config-if)# description CONNECTS to CE5
Router(config-if)# no ip address
Router(config-if)# no keepalive
Router(config-if)# xconnect 172.16.255.255 100 encapsulation mpls manual pw-class static-pw
Router(config-if-xconn)# mpls label 16 17
Router(config-if-xconn)# mpls control-word
!
```

Example Configuring the Tailend Router (PE1)

In the following sample configuration of the tailend router, note the following:

- All the tailend routers must use the same binding configuration.
- The **xconnect** command must always be configured on tailend routers.

```
Router(config)# pseudowire-class static-pw
Router(config-pw-class)# encapsulation mpls
Router(config-pw-class)# protocol none
!
Router(config)# interface loopback 0
Router(config-if)# ip address 172.16.255.1 255.255.255.255
!
Router(config)# interface ethernet 0/0
Router(config-if)# description CONNECTS TO CE1
Router(config-if)# no ip address
Router(config-if)# no keepalive
Router(config-if)# xconnect 172.16.255.5 100 encapsulation mpls manual pw-class static-pw
Router(config-if-xconn)# mpls label 17 16
Router(config-if-xconn)# mpls control-word
!
```

CHAPTER 6

MPLS Traffic Engineering – Bundled Interface Support

Note This technology is not applicable for the Cisco ASR 900 RSP3 Module.

The MPLS Traffic Engineering - Bundled Interface Support feature enables Multiprotocol Label Switching (MPLS) traffic engineering (TE) tunnels over the bundled interfaces—EtherChannel and Gigabit EtherChannel (GEC).

The Resource Reservation Protocol (RSVP) notifies TE about bandwidth changes that occur when member links are added or deleted, or when links become active or inactive. TE notifies other nodes in the network via Interior Gateway Protocol (IGP) flooding. By default, the bandwidth available to TE Label-Switched Paths (LSPs) is 75 percent of the interface bandwidth. You can change the percentage of the global bandwidth available for TE LSPs by using an RSVP command on the bundled interface. Bandwidth reservation and preemption are supported.

The Fast Reroute (FRR) feature is supported on bundled interfaces. FRR is activated when a bundled interface goes down; for example, if you enter the **shutdown** command to shut down the interface or fewer than the required minimum number of links are operational.

Prerequisites for MPLS TE – Bundled Interface Support

- Configure Multiprotocol Label Switching (MPLS) traffic engineering (TE) tunnels.
- Enable Cisco Express Forwarding in global configuration mode.
- Enable Resource Reservation Protocol (RSVP) feature.
- Configure EtherChannel.
- Configure Gigabit EtherChannel.

Restrictions for MPLS TE – Bundled Interface Support

- Traffic engineering over switch virtual interfaces (SVIs) is not supported unless the SVI consists of a bundle of links that represent a single point-to-point interface.
- There must be a valid IP address configuration on the bundled interface and there must not be an IP address configuration on the member links.

Information About MPLS TE – Bundled Interface Support

Cisco EtherChannel Overview

Cisco EtherChannel technology builds upon standards-based 802.3 full-duplex Fast Ethernet to provide network managers with a reliable, high-speed solution for the campus network backbone. EtherChannel technology provides bandwidth scalability within the campus by providing up to 800 Mbps, 8 Gbps, or 80 Gbps of aggregate bandwidth for a Fast EtherChannel, Gigabit EtherChannel, or 10 Gigabit EtherChannel connection, respectively. Each of these connection speeds can vary in amounts equal to the speed of the links used (100 Mbps, 1 Gbps, or 10 Gbps). Even in the most bandwidth-demanding situations, EtherChannel technology helps to aggregate traffic, keeps oversubscription to a minimum, and provides effective link-resiliency mechanisms.

Cisco EtherChannel Benefits

Cisco EtherChannel technology allows network managers to provide higher bandwidth among servers, routers, and switches than a single-link Ethernet technology can provide.

Cisco EtherChannel technology provides incremental scalable bandwidth and the following benefits:

- Standards-based—Cisco EtherChannel technology builds upon IEEE 802.3-compliant Ethernet by grouping multiple, full-duplex point-to-point links. EtherChannel technology uses IEEE 802.3 mechanisms for full-duplex autonegotiation and autosensing, when applicable.
- Flexible incremental bandwidth—Cisco EtherChannel technology provides bandwidth aggregation in multiples of 100 Mbps, 1 Gbps, or 10 Gbps, depending on the speed of the aggregated links. For example, network managers can deploy EtherChannel technology that consists of pairs of full-duplex Fast Ethernet links to provide more than 400 Mbps between the wiring closet and the data center. In the data center, bandwidths of up to 800 Mbps can be provided between servers and the network backbone to provide large amounts of scalable incremental bandwidth.
- Load balancing—Cisco EtherChannel technology comprises several Fast Ethernet links and is capable of load balancing traffic across those links. Unicast, broadcast, and multicast traffic is evenly distributed across the links, providing improved performance and redundant parallel paths. When a link fails, traffic is redirected to the remaining links within the channel without user intervention and with minimal packet loss.
- Resiliency and fast convergence—When a link fails, Cisco EtherChannel technology provides automatic recovery by redistributing the load across the remaining links. When a link fails, Cisco EtherChannel technology redirects traffic from the failed link to the remaining links in less than one second. This convergence is transparent to the end user—no host protocol timers expire and no sessions are dropped.

Cisco Gigabit EtherChannel Overview

Cisco Gigabit EtherChannel (GEC) is a high-performance Ethernet technology that provides transmission rates in Gigabit per second (Gbps). A Gigabit EtherChannel bundles individual ethernet links (Gigabit Ethernet and 10 Gigabit Ethernet) into a single logical link that provides the aggregate bandwidth up to four physical links. All LAN ports in each EtherChannel must be of the same speed and must be configured as either Layer 2 or Layer 3 LAN ports. Inbound broadcast and multicast packets on one link in an EtherChannel are blocked from returning on any other link in the EtherChannel.

Load Balancing and Min-Links in EtherChannel

Load balancing affects the actual and practical bandwidth that can be used for TE. Multilink load balancing uses a per-packet load balancing method. All of the bundle interface bandwidth is available. EtherChannel load balancing has various load balancing methods, depending on the traffic pattern and the load balancing configuration. The total bandwidth available for TE may be limited to the bandwidth of a single member link.

On EtherChannel, min-links is supported only in the Link Aggregation Control Protocol (LACP). For other EtherChannel protocols, the minimum is one link, by default, and it is not configurable. To configure min-links for EtherChannel, use the **port-channel min-links** command.

How to Configure MPLS TE – Bundled Interface Support

Configuring MPLS TE on an EtherChannel Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number* [*name-tag*]
4. **ip address** *ip-address mask* [**secondary**]
5. **mpls traffic-eng tunnels**
6. **mpls traffic-eng backup-path** *tunnel*
7. **port-channel min-links** *min-num*
8. **ip rsvp bandwidth** [*interface-kbps*] [*single-flow-kbps*]
9. **end**

DETAILED STEPS

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

	Command or Action	Purpose
	`Device# configure terminal`	
Step 3	**interface** *type number* [*name-tag*] **Example:** `Device(config)# interface port-channel 1`	Creates an EtherChannel bundle, assigns a group number to the bundle, and enters interface configuration mode.
Step 4	**ip address** *ip-address mask* [**secondary**] **Example:** `Device(config-if)# ip address 10.0.0.4 255.255.255.0`	Specifies an IP address for the EtherChannel group.
Step 5	**mpls traffic-eng tunnels** **Example:** `Device(config-if)# mpls traffic-eng tunnels`	Enables MPLS TE tunnel signaling on an interface. • MPLS TE tunnel should be enabled on the device before enabling the signaling.
Step 6	**mpls traffic-eng backup-path** *tunnel* **Example:** `Device(config-if)# mpls traffic-eng backup-path Tunnel120`	(Optional) Configures the physical interface to use a backup tunnel in the event of a detected failure on that interface.
Step 7	**port-channel min-links** *min-num* **Example:** `Device(config-if)# port-channel min-links 2`	Specifies that a minimum number of bundled ports in an EtherChannel is required before the channel can be active.
Step 8	**ip rsvp bandwidth** [*interface-kbps*] [*single-flow-kbps*] **Example:** `Device(config-if)# ip rsvp bandwidth 100`	Enables RSVP for IP on an interface and specifies a percentage of the total interface bandwidth as available in the RSVP bandwidth pool.

Step 9	**end**
	Example:
	`Device(config-if)# end`
		Exits interface configuration mode and returns to privileged EXEC mode.

Configuration Examples for MPLS TE Bundled Interface Support

Example: Configuring MPLS TE on an EtherChannel Interface

```
Device> enable
```

```
Device# configure terminal
Device(config)# interface port-channel 1
Device(config-if)# ip address 10.0.0.4 255.255.255.0
Device(config-if)# mpls traffic-eng tunnels
Device(config-if)# mpls traffic-eng backup-path Tunnel 120
Device(config-if)# port-channel min-links 2
Device(config-if)# ip rsvp bandwidth 100
Device(config-if)# end
```

Example: Configuring MPLS TE - Bundled Interface Support over Gigabit Etherchannel

The following example shows how to enable MPLS TE – bundled interface support over GEC on Cisco devices:

```
Device> enable
Device# configure terminal

! Enable global MPLS TE on routers
Device(config)# router ospf 100
Device(config-router)# network 10.0.0.1 0.0.0.255 area 0
Device(config-router)# mpls traffic-eng area 0
Device(config-router)# mpls traffic-eng router-id Loopback 0
Device(config-router)# exit

! Configure  GEC  interface and enable MPLS TE and RSVP on interface
Device(config)# interface Port-channel 1
Device(config-if)# ip address 10.0.0.1 255.255.255.0
Device(config-if)# mpls traffic-eng tunnels
Device(config-if)# ip rsvp bandwidth
Device(config-if)# exit

! Define explicit path
Device(config)# ip explicit-path name primary enable
Device(cfg-ip-expl-path)# next-address 172.12.1.2
Device(cfg-ip-expl-path)# next-address 172.23.1.2
Device(cfg-ip-expl-path)# next-address 172.34.1.2
Device(cfg-ip-expl-path)# next-address 10.4.4.4
Device(cfg-ip-expl-path)# exit

! Configure primary tunnel on head-end device
Device(config)# interface Tunnel 14
Device(config-if)# ip unnumbered Loopback 0
Device(config-if)# tunnel mode mpls traffic-eng
Device(config-if)# tunnel destination 10.10.10.0
Device(config-if)# tunnel mpls traffic-eng autoroute announce
Device(config-if)# tunnel mpls traffic-eng path-option 10 explicit name primary
Device(config-if)# tunnel mpls traffic-eng fast-reroute
Device(config-if)# exit

! Configure backup tunnel on head-end or mid-point device
Device(config)# interface Tunnel 23
Device(config-if)# ip unnumbered Looback 0
Device(config-if)# tunnel mode mpls traffic-eng
Device(config-if)# tunnel destination 10.20.10.0
Device(config-if)# tunnel mpls traffic-eng path-option 10 explicit name backup
Device(config-if)# exit
```

```
! Configure backup tunnel on protected GEC interface
Device(config)# interface Port-channel 1
Device(config-if)# ip address 10.0.0.1 255.255.255.0
Device(config-if)# mpls traffic-eng tunnels
Device(config-if)# mpls traffic-eng backup-path Tunnel 23
Device(config-if)# ip rsvp bandwidth percent 20
Device(config-if)# lacp min-bundle 2
Device(config-if)# exit

! Configure GEC interface
Device(config)# interface GigabitEthernet 0/0/1
Device(config-if)# no ip address
Device(config-if)# channel-group 1 mode active
Device(config-if)# exit

! Configure GEC interface
Device(config)# interface GigabitEthernet 0/0/2
Device(config-if)# no ip address
Device(config-if)# channel-group 1 mode active
Device(config-if)# exit
```

The **show mpls traffic-eng tunnels** command output displays information about a tunnel or one–line information about all tunnels configured on the device:

```
Device# show mpls traffic-eng tunnels tunnel 14

Name: ASR1013_t14                         (Tunnel10) Destination: 10.4.4.4
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 1, type explicit toR4overR3R3 (Basis for Setup, path weight 3)

  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 7  7   Affinity: 0x0/0xFFFF
    Metric Type: TE (default)
    AutoRoute: enabled  LockDown: disabled Loadshare: 0 [0] bw-based
    auto-bw: disabled
  Active Path Option Parameters:
    State: explicit path option 1 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled


      InLabel  :  -
  OutLabel : Port-channel1, 1608
  Next Hop : 172.16.1.2
  FRR OutLabel : Tunnel23, 4868
  RSVP Signalling Info:
       Src 10.1.1.1, Dst 10.4.4.4, Tun_Id 14, Tun_Instance 35
    RSVP Path Info:
      My Address: 172.12.1.1
      Explicit Route: 172.12.1.2 172.23.1.1 172.23.1.2 172.34.1.1
                      172.34.1.2 10.4.4.4

  History:
    Tunnel:
      Time since created: 17 hours
      Time since path change: 18 minutes, 22 seconds
      Number of LSP IDs (Tun_Instances) used: 35
    Current LSP: [ID: 35]
      Uptime: 18 minutes, 22 seconds
      Selection: reoptimization
    Prior LSP: [ID: 32]
```

```
      ID: path option unknown
      Removal Trigger: signalling shutdown

Device# show mpls traffic-eng tunnels brief

show mpls traffic-eng tunnels brief
Signalling Summary:
    LSP Tunnels Process:            running
    Passive LSP Listener:           running
    RSVP Process:                   running
    Forwarding:                     enabled
    Periodic reoptimization:        every 3600 seconds, next in 3299 seconds
    Periodic FRR Promotion:         Not Running
    Periodic auto-bw collection:    every 300 seconds, next in 299 seconds

P2P TUNNELS/LSPs:
TUNNEL NAME                      DESTINATION      UP IF     DOWN IF   STATE/PROT^M
ASR1013_t14                      10.4.1.1                       -         Po12      up/up
On Mid Router:
P2P TUNNELS/LSPs:
TUNNEL NAME                      DESTINATION      UP IF     DOWN IF   STATE/PROT
ASR1013_t14                       10.4.1.1                   Po12       Po23       up/up
ASR1002F_t23                      10.2.1.1                   Po25        -         up/up
```

The **show mpls traffic-eng fast-reroute** command output displays information about FRR-protected MPLS TE tunnels originating, transmitting, or terminating on this device.

```
Device# show mpls traffic-eng fast-reroute database

P2P Headend FRR information:
Protected tunnel               In-label Out intf/label   FRR intf/label   Status
---------------------------    -------- --------------   --------------   ------

P2P LSP midpoint frr information:
LSP identifier                 In-label Out intf/label   FRR intf/label   Status
---------------------------    -------- --------------   --------------   ------
10.1.1.1 1 [2]                 16        Po23:16          Tu23:16          active
```