Implementing MPLS Traffic Engineering

This module describes how to implement MPLS Traffic Engineering on Cisco ASR 9000 Series Router.

Multiprotocol Label Switching (MPLS) is a standards-based solution driven by the Internet Engineering Task Force (IETF) that was devised to convert the Internet and IP backbones from best-effort networks into business-class transport mediums.

MPLS, with its label switching capabilities, eliminates the need for an IP route look-up and creates a virtual circuit (VC) switching function, allowing enterprises the same performance on their IP-based network services as with those delivered over traditional networks such as Frame Relay or Asynchronous Transfer Mode (ATM).

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

Feature History for Implementing MPLS-TE

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 3.7.2</td>
<td>This feature was introduced.</td>
</tr>
<tr>
<td>Release 3.9.0</td>
<td>The MPLS Traffic Engineering (TE): Path Protection feature was added.</td>
</tr>
<tr>
<td>Release 3.9.1</td>
<td>The MPLS-TE automatic bandwidth feature is supported.</td>
</tr>
<tr>
<td>Release 4.0.0</td>
<td>Support was added for the following features:</td>
</tr>
<tr>
<td></td>
<td>• AutoTunnel Backup</td>
</tr>
<tr>
<td></td>
<td>• SRLG (Shared Risk Link Groups)</td>
</tr>
<tr>
<td>Release 4.1.0</td>
<td>Support was added for the following features:</td>
</tr>
<tr>
<td></td>
<td>• Ignore Intermediate System-to-Intermediate System Overload Bit Setting in MPLS-TE</td>
</tr>
<tr>
<td></td>
<td>• Point-to-Multipoint Traffic-Engineering</td>
</tr>
</tbody>
</table>
Prerequisites for Implementing Cisco MPLS Traffic Engineering

These prerequisites are required to implement MPLS TE:

• You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

• Router that runs Cisco IOS XR software.

• Installed composite mini-image and the MPLS package, or a full composite image.

• IGP activated.

Restrictions for Implementing Cisco MPLS Traffic Engineering

In addition to the MPLS-TE Fast Reroute feature supporting the GigabitEthernet and TenGigE line cards, this current release also supports the 8-port OC-12 SPA, 2-port OC-48 SPA, 1-port OC-192 SPA, along with the Cisco ASR 9000 Series SPA Interface Processor-700. This feature is also supported on the main interfaces on the SPA line cards, not on sub-interfaces. There is no support for the MPLS-TE Fast Reroute feature on the 2-port channelized OC-12 SPA or on the 1-port channelized OC-48 SPA.

Information About Implementing MPLS Traffic Engineering

To implement MPLS-TE, you should understand these concepts:

Overview of MPLS Traffic Engineering

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

MPLS-TE is essential for service provider and Internet service provider (ISP) backbones. Such backbones must support a high use of transmission capacity, and the networks must be very resilient so that they can withstand link or node failures. MPLS-TE provides an integrated approach to traffic engineering. With MPLS, traffic engineering capabilities are integrated into Layer 3, which optimizes the routing of IP traffic, given the constraints imposed by backbone capacity and topology.
Benefits of MPLS Traffic Engineering

MPLS-TE enables ISPs to route network traffic to offer the best service to their users in terms of throughput and delay. By making the service provider more efficient, traffic engineering reduces the cost of the network.

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

How MPLS-TE Works

MPLS-TE automatically establishes and maintains label switched paths (LSPs) across the backbone by using RSVP. The path that an LSP uses is determined by the LSP resource requirements and network resources, such as bandwidth. Available resources are flooded by means of extensions to a link-state-based Interior Gateway Protocol (IGP).

MPLS-TE tunnels are calculated at the LSP headend router, based on a fit between the required and available resources (constraint-based routing). The IGP automatically routes the traffic to these LSPs.

Typically, a packet crossing the MPLS-TE backbone travels on a single LSP that connects the ingress point to the egress point. MPLS-TE is built on these mechanisms:

- **Tunnel interfaces**: From a Layer 2 standpoint, an MPLS tunnel interface represents the headend of an LSP. It is configured with a set of resource requirements, such as bandwidth and media requirements, and priority. From a Layer 3 standpoint, an LSP tunnel interface is the headend of a unidirectional virtual link to the tunnel destination.

- **MPLS-TE path calculation module**: This calculation module operates at the LSP headend. The module determines a path to use for an LSP. The path calculation uses a link-state database containing flooded topology and resource information.

- **RSVP with TE extensions**: RSVP operates at each LSP hop and is used to signal and maintain LSPs based on the calculated path.

- **MPLS-TE link management module**: This module operates at each LSP hop, performs link call admission on the RSVP signaling messages, and performs bookkeeping on topology and resource information to be flooded.

- **Link-state IGP (Intermediate System-to-Intermediate System [IS-IS] or Open Shortest Path First [OSPF]—each with traffic engineering extensions)**: These IGPs are used to globally flood topology and resource information from the link management module.
Enhancements to the shortest path first (SPF) calculation used by the link-state IGP (IS-IS or OSPF)

The IGP automatically routes traffic to the appropriate LSP tunnel, based on tunnel destination. Static routes can also be used to direct traffic to LSP tunnels.

Label switching forwarding

This forwarding mechanism provides routers with a Layer 2-like ability to direct traffic across multiple hops of the LSP established by RSVP signaling.

One approach to engineering a backbone is to define a mesh of tunnels from every ingress device to every egress device. The MPLS-TE path calculation and signaling modules determine the path taken by the LSPs for these tunnels, subject to resource availability and the dynamic state of the network.

The IGP (operating at an ingress device) determines which traffic should go to which egress device, and steers that traffic into the tunnel from ingress to egress. A flow from an ingress device to an egress device might be so large that it cannot fit over a single link, so it cannot be carried by a single tunnel. In this case, multiple tunnels between a given ingress and egress can be configured, and the flow is distributed using load sharing among the tunnels.

Related Topics

- Building MPLS-TE Topology, page 30
- Creating an MPLS-TE Tunnel, page 33
- Build MPLS-TE Topology and Tunnels: Example, page 115

MPLS Traffic Engineering

Multiprotocol Label Switching (MPLS) is an Internet Engineering Task Force (IETF)-specified framework that provides efficient designation, routing, forwarding, and switching of traffic flows through the network.

TE is the process of adjusting bandwidth allocations to ensure that enough bandwidth is available for high-priority traffic.

In MPLS TE, the upstream router creates a network tunnel for a particular traffic stream and sets the bandwidth available for that tunnel.

Backup AutoTunnels

The MPLS Traffic Engineering AutoTunnel Backup feature enables a router to dynamically build backup tunnels on the interfaces that are configured with MPLS TE tunnels. This feature enables a router to dynamically build backup tunnels when they are needed. This prevents you from having to build MPLS TE tunnels statically.

The MPLS Traffic Engineering (TE) AutoTunnel Backup feature has these benefits:

- Backup tunnels are built automatically, eliminating the need for users to preconfigure each backup tunnel and then assign the backup tunnel to the protected interface.
- Protection is expanded—FRR does not protect IP traffic that is not using the TE tunnel or Label Distribution Protocol (LDP) labels that are not using the TE tunnel.

This feature protects against these failures:

- **P2P Tunnel NHOP protection**—Protects against link failure for the associated P2P protected tunnel
- **P2P Tunnel NNHOP protection**—Protects against node failure for the associated P2P protected tunnel
P2MP Tunnel NHOP protection—Protects against link failure for the associated P2MP protected tunnel.

Related Topics
- Enabling an AutoTunnel Backup, page 43
- Removing an AutoTunnel Backup, page 44
- Establishing MPLS Backup AutoTunnels to Protect Fast Reroutable TE LSPs, page 46
- Establishing Next-Hop Tunnels with Link Protection, page 47
- Configure the MPLS-TE Auto-Tunnel Backup: Example, page 123

Link Protection

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

Link Protection, page 5 illustrates link protection.

Figure 1: Link Protection

Node Protection

The backup tunnels that bypass next-hop nodes along LSP paths are called NNHOP backup tunnels because they terminate at the node following the next-hop node of the LSPs, thereby bypassing the next-hop node. They protect LSPs by enabling the node upstream of a link or node failure to reroute the LSPs and their traffic around a node failure to the next-hop node. NNHOP backup tunnels also provide protection from link failures because they bypass the failed link and the node.
Backup AutoTunnel Assignment

At the head or mid points of a tunnel, the backup assignment finds an appropriate backup to protect a given primary tunnel for FRR protection.

The backup assignment logic is performed differently based on the type of backup configured on the output interface used by the primary tunnel. Configured backup types are:

- Static Backup
- AutoTunnel Backup
- No Backup (In this case no backup assignment is performed and the tunnels is unprotected.)

Note: Static backup and Backup AutoTunnel cannot exist together on the same interface or link.

Note: Node protection is always preferred over link protection in the Backup AutoTunnel assignment.

In order that the Backup AutoTunnel feature operates successfully, the following configuration must be applied at global configuration level:

```
ipv4 unnumbered mpls traffic-eng Loopback 0
```

Note: The Loopback 0 is used as router ID.
Explicit Paths

Explicit paths are used to create backup autotunnels as follows:

For NHOP Backup Autotunnels:

• NHOP excludes the protected link’s local IP address.
• NHOP excludes the protected link’s remote IP address.
• The explicit-path name is _autob_nhoptunnelxxx, where xxx matches the dynamically created backup tunnel ID.

For NNHOP Backup Autotunnels:

• NNHOP excludes the protected link’s local IP address.
• NNHOP excludes the protected link’s remote IP address (link address on next hop).
• NNHOP excludes the NHOP router ID of the protected primary tunnel next hop.
• The explicit-path name is _autob_nhoptunnelxxx, where xxx matches the dynamically created backup tunnel ID.

Periodic Backup Promotion

The periodic backup promotion attempts to find and assign a better backup for primary tunnels that are already protected.

With AutoTunnel Backup, the only scenario where two backups can protect the same primary tunnel is when both an NHOP and NNHOP AutoTunnel Backups get created. The backup assignment takes place as soon as the NHOP and NNHOP backup tunnels come up. So, there is no need to wait for the periodic promotion.

Although there is no exception for AutoTunnel Backups, periodic backup promotion has no impact on primary tunnels protected by AutoTunnel Backup.

One exception is when a manual promotion is triggered by the user using the mpls traffic-eng fast-reroute timers promotion command, where backup assignment or promotion is triggered on all FRR protected primary tunnels— even unprotected ones. This may trigger the immediate creation of some AutoTunnel Backup, if the command is entered within the time window when a required AutoTunnel Backup has not been yet created.

You can configure the periodic promotion timer using the global configuration mpls traffic-eng fast-reroute timers promotion sec command. The range is 0 to 604800 seconds.

Note

A value of 0 for the periodic promotion timer disables the periodic promotion.

Protocol-Based CLI

Cisco IOS XR software provides a protocol-based command line interface. The CLI provides commands that can be used with the multiple IGP protocols supported by MPLS-TE.
Differentiated Services Traffic Engineering

MPLS Differentiated Services (Diff-Serv) Aware Traffic Engineering (DS-TE) is an extension of the regular MPLS-TE feature. Regular traffic engineering does not provide bandwidth guarantees to different traffic classes. A single bandwidth constraint is used in regular TE that is shared by all traffic. To support various classes of service (CoS), users can configure multiple bandwidth constraints. These bandwidth constraints can be treated differently based on the requirement for the traffic class using that constraint.

MPLS DS-TE provides the ability to configure multiple bandwidth constraints on an MPLS-enabled interface. Available bandwidths from all configured bandwidth constraints are advertised using IGP. TE tunnel is configured with bandwidth value and class-type requirements. Path calculation and admission control take the bandwidth and class-type into consideration. RSVP is used to signal the TE tunnel with bandwidth and class-type requirements.

MPLS DS-TE is deployed with either Russian Doll Model (RDM) or Maximum Allocation Model (MAM) for bandwidth calculations.

Cisco IOS XR software supports two DS-TE modes: Prestandard and IETF.

Related Topics
- Confirming DiffServ-TE Bandwidth
- Bandwidth Configuration (MAM): Example
- Bandwidth Configuration (RDM): Example

Prestandard DS-TE Mode

Prestandard DS-TE uses the Cisco proprietary mechanisms for RSVP signaling and IGP advertisements. This DS-TE mode does not interoperate with third-party vendor equipment. Note that prestandard DS-TE is enabled only after configuring the sub-pool bandwidth values on MPLS-enabled interfaces.

Prestandard Diff-Serve TE mode supports a single bandwidth constraint model a Russian Doll Model (RDM) with two bandwidth pools: global-pool and sub-pool.

TE class map is not used with Prestandard DS-TE mode.

Related Topics
- Configuring a Prestandard DS-TE Tunnel, page 49
- Configure IETF DS-TE Tunnels: Example, page 116

IETF DS-TE Mode

IETF DS-TE mode uses IETF-defined extensions for RSVP and IGP. This mode interoperates with third-party vendor equipment.

IETF mode supports multiple bandwidth constraint models, including RDM and MAM, both with two bandwidth pools. In an IETF DS-TE network, identical bandwidth constraint models must be configured on all nodes.

TE class map is used with IETF DS-TE mode and must be configured the same way on all nodes in the network.
Bandwidth Constraint Models

IETF DS-TE mode provides support for the RDM and MAM bandwidth constraints models. Both models support up to two bandwidth pools.

Cisco IOS XR software provides global configuration for the switching between bandwidth constraint models. Both models can be configured on a single interface to preconfigure the bandwidth constraints before swapping to an alternate bandwidth constraint model.

Note

NSF is not guaranteed when you change the bandwidth constraint model or configuration information.

By default, RDM is the default bandwidth constraint model used in both pre-standard and IETF mode.

Maximum Allocation Bandwidth Constraint Model

The MAM constraint model has the following characteristics:

• Easy to use and intuitive.
• Isolation across class types.
• Simultaneously achieves isolation, bandwidth efficiency, and protection against QoS degradation.

Related Topics

• Configuring an IETF DS-TE Tunnel Using MAM, page 54

Russian Doll Bandwidth Constraint Model

The RDM constraint model has these characteristics:

• Allows greater sharing of bandwidth among different class types.
• Ensures bandwidth efficiency simultaneously and protection against QoS degradation of all class types.
• Specifies that it is used in conjunction with preemption to simultaneously achieve isolation across class-types such that each class-type is guaranteed its share of bandwidth, bandwidth efficiency, and protection against QoS degradation of all class types.

Note

We recommend that RDM not be used in DS-TE environments in which the use of preemption is precluded. Although RDM ensures bandwidth efficiency and protection against QoS degradation of class types, it does guarantee isolation across class types.

Related Topics

• Configuring an IETF DS-TE Tunnel Using RDM, page 52
**TE Class Mapping**

Each of the eight available bandwidth values advertised in the IGP corresponds to a TE class. Because the IGP advertises only eight bandwidth values, there can be a maximum of only eight TE classes supported in an IETF DS-TE network.

TE class mapping must be exactly the same on all routers in a DS-TE domain. It is the responsibility of the operator configure these settings properly as there is no way to automatically check or enforce consistency.

The operator must configure TE tunnel class types and priority levels to form a valid TE class. When the TE class map configuration is changed, tunnels already up are brought down. Tunnels in the down state, can be set up if a valid TE class map is found.

The default TE class and attributes are listed. The default mapping includes four class types.

**Table 1: TE Classes and Priority**

<table>
<thead>
<tr>
<th>TE Class</th>
<th>Class Type</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>7</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>Unused</td>
<td>—</td>
</tr>
<tr>
<td>3</td>
<td>Unused</td>
<td>—</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>Unused</td>
<td>—</td>
</tr>
<tr>
<td>7</td>
<td>Unused</td>
<td>—</td>
</tr>
</tbody>
</table>

**Flooding**

Available bandwidth in all configured bandwidth pools is flooded on the network to calculate accurate constraint paths when a new TE tunnel is configured. Flooding uses IGP protocol extensions and mechanisms to determine when to flood the network with bandwidth.

**Flooding Triggers**

TE Link Management (TE-Link) notifies IGP for both global pool and sub-pool available bandwidth and maximum bandwidth to flood the network in these events:

- Periodic timer expires (this does not depend on bandwidth pool type).
- Tunnel origination node has out-of-date information for either available global pool or sub-pool bandwidth, causing tunnel admission failure at the midpoint.
• Consumed bandwidth crosses user-configured thresholds. The same threshold is used for both global pool and sub-pool. If one bandwidth crosses the threshold, both bandwidths are flooded.

Flooding Thresholds

Flooding frequently can burden a network because all routers must send out and process these updates. Infrequent flooding causes tunnel heads (tunnel-originating nodes) to have out-of-date information, causing tunnel admission to fail at the midpoints.

You can control the frequency of flooding by configuring a set of thresholds. When locked bandwidth (at one or more priority levels) crosses one of these thresholds, flooding is triggered.

Thresholds apply to a percentage of the maximum available bandwidth (the global pool), which is locked, and the percentage of maximum available guaranteed bandwidth (the sub-pool), which is locked. If, for one or more priority levels, either of these percentages crosses a threshold, flooding is triggered.

Note Setting up a global pool TE tunnel can cause the locked bandwidth allocated to sub-pool tunnels to be reduced (and hence to cross a threshold). A sub-pool TE tunnel setup can similarly cause the locked bandwidth for global pool TE tunnels to cross a threshold. Thus, sub-pool TE and global pool TE tunnels can affect each other when flooding is triggered by thresholds.

Fast Reroute

Fast Reroute (FRR) provides link protection to LSPs enabling the traffic carried by LSPs that encounter a failed link to be rerouted around the failure. The reroute decision is controlled locally by the router connected to the failed link. The headend router on the tunnel is notified of the link failure through IGP or through RSVP. When it is notified of a link failure, the headend router attempts to establish a new LSP that bypasses the failure. This provides a path to reestablish links that fail, providing protection to data transfer.

FRR (link or node) is supported over sub-pool tunnels the same way as for regular TE tunnels. In particular, when link protection is activated for a given link, TE tunnels eligible for FRR are redirected into the protection LSP, regardless of whether they are sub-pool or global pool tunnels.

Note The ability to configure FRR on a per-LSP basis makes it possible to provide different levels of fast restoration to tunnels from different bandwidth pools.

You should be aware of these requirements for the backup tunnel path:

• Backup tunnel must not pass through the element it protects.

• Primary tunnel and a backup tunnel should intersect at least at two points (nodes) on the path: point of local repair (PLR) and merge point (MP). PLR is the headend of the backup tunnel, and MP is the tailend of the backup tunnel.

Note When you configure TE tunnel with multiple protection on its path and merge point is the same node for more than one protection, you must configure record-route for that tunnel.
Related Topics

- Protecting MPLS Tunnels with Fast Reroute, page 38

MPLS-TE and Fast Reroute over Link Bundles

MPLS Traffic Engineering (TE) and Fast Reroute (FRR) are supported over bundle interfaces and virtual local area network (VLAN) interfaces. Bidirectional forwarding detection (BFD) over VLAN is used as an FRR trigger to obtain less than 50 milliseconds of switchover time.

These link bundle types are supported for MPLS-TE/FRR:

- Over Ethernet link bundles.
- Over VLANs over Ethernet link bundles.
- Number of links are limited to 100 for MPLS-TE and FRR.
- VLANs go over any Ethernet interface (for example, GigabitEthernet and TenGigE).

FRR is supported over bundle interfaces in the following ways:

- Uses minimum links as a threshold to trigger FRR over a bundle interface.
- Uses the minimum total available bandwidth as a threshold to trigger FRR.

Ignore Intermediate System-to-Intermediate System Overload Bit Setting in MPLS-TE

The Ignore Intermediate System-to-Intermediate System (IS-IS) overload bit avoidance feature allows network administrators to prevent RSVP-TE label switched paths (LSPs) from being disabled, when a router in that path has its Intermediate System-to-Intermediate System (IS-IS) overload bit set.

The IS-IS overload bit avoidance feature is activated using this command:

```
mpls traffic-eng path-selection ignore overload
```

The IS-IS overload bit avoidance feature is deactivated using the `no` form of this command:

```
no mpls traffic-eng path-selection ignore overload
```

When the IS-IS overload bit avoidance feature is activated, all nodes, including head nodes, mid nodes, and tail nodes, with the overload bit set, are ignored. This means that they are still available for use with RSVP-TE label switched paths (LSPs). This feature enables you to include an overloaded node in CSPF.

Enhancement Options of IS-IS OLA

You can restrict configuring IS-IS overload bit avoidance with the following enhancement options:

- **path-selection ignore overload head**
  The tunnels stay up if `set-overload-bit` is set by IS-IS on the head router. Ignores overload during CSPF for LSPs originating from an overloaded node. In all other cases (mid, tail, or both), the tunnel stays down.

- **path-selection ignore overload mid**
The tunnels stay up if set-overload-bit is set by IS-IS on the mid router. Ignores overload during CSPF for LSPs transiting from an overloaded node. In all other cases (head, tail, or both), the tunnel stays down.

• path-selection ignore overload tail

The tunnels stay up if set-overload-bit is set by IS-IS on the tail router. Ignores overload during CSPF for LSPs terminating at an overloaded node. In all other cases (head, mid, or both), the tunnel stays down.

• path-selection ignore overload

The tunnels stay up irrespective of on which router the set-overload-bit is set by IS-IS.

Note

When you do not select any of the options, including head nodes, mid nodes, and tail nodes, you get a behavior that is applicable to all nodes. This behavior is backward compatible in nature.

For more information related to IS-IS overload avoidance related commands, see Cisco ASR 9000 Series Aggregation Services Router MPLS Command Reference.

Related Topics

• Configuring the Ignore Integrated IS-IS Overload Bit Setting in MPLS-TE, page 59
• Configure the Ignore IS-IS Overload Bit Setting in MPLS-TE: Example, page 117

Flexible Name-based Tunnel Constraints

MPLS-TE Flexible Name-based Tunnel Constraints provides a simplified and more flexible means of configuring link attributes and path affinities to compute paths for MPLS-TE tunnels.

In the traditional TE scheme, links are configured with attribute-flags that are flooded with TE link-state parameters using Interior Gateway Protocols (IGPs), such as Open Shortest Path First (OSPF).

MPLS-TE Flexible Name-based Tunnel Constraints lets you assign, or map, up to 32 color names for affinity and attribute-flag attributes instead of 32-bit hexadecimal numbers. After mappings are defined, the attributes can be referred to by the corresponding color name in the command-line interface (CLI). Furthermore, you can define constraints using include, include-strict, exclude, and exclude-all arguments, where each statement can contain up to 10 colors, and define include constraints in both loose and strict sense.

Note

You can configure affinity constraints using attribute flags or the Flexible Name Based Tunnel Constraints scheme; however, when configurations for both schemes exist, only the configuration pertaining to the new scheme is applied.

Related Topics

• Assigning Color Names to Numeric Values, page 61
• Associating Affinity-Names with TE Links, page 63
• Associating Affinity Constraints for TE Tunnels, page 64
• Configure Flexible Name-based Tunnel Constraints: Example, page 117
MPLS Traffic Engineering Interarea Tunneling

These topics describe the following new extensions of MPLS-TE:

- Interarea Support, page 14
- Multiarea Support, page 14
- Loose Hop Expansion, page 15
- Loose Hop Reoptimization, page 15
- Fast Reroute Node Protection, page 16

Interarea Support

The MPLS-TE interarea tunneling feature allows you to establish TE tunnels spanning multiple Interior Gateway Protocol (IGP) areas and levels, thereby eliminating the requirement that headend and tailend routers reside in a single area.

Interarea support allows the configuration of a TE LSP that spans multiple areas, where its headend and tailend label switched routers (LSRs) reside in different IGP areas.

Multiarea and Interarea TE are required by the customers running multiple IGP area backbones (primarily for scalability reasons). This lets you limit the amount of flooded information, reduces the SPF duration, and lessens the impact of a link or node failure within an area, particularly with large WAN backbones split in multiple areas.

This figure shows a typical interarea TE network.

Figure 3: Interarea (OSPF) TE Network Diagram

Multiarea Support

Multiarea support allows an ABR LSR to support MPLS-TE in more than one IGP area. A TE LSP is still confined to a single area.

Multiarea and Interarea TE are required when you run multiple IGP area backbones. The Multiarea and Interarea TE allows you to:

- Limit the volume of flooded information.
• Reduce the SPF duration.
• Decrease the impact of a link or node failure within an area.

**Figure 4: Interlevel (IS-IS) TE Network**

As shown in the figure, R2, R3, R7, and R4 maintain two databases for routing and TE information. For example, R3 has TE topology information related to R2, flooded through Level-1 IS-IS LSPs plus the TE topology information related to R4, R9, and R7, flooded as Level 2 IS-IS Link State PDUs (LSPs) (plus, its own IS-IS LSP).

---

**Note**

You can configure multiple areas within an IS-IS Level 1. This is transparent to TE. TE has topology information about the IS-IS level, but not the area ID.

---

**Loose Hop Expansion**

Loose hop optimization allows the reoptimization of tunnels spanning multiple areas and solves the problem which occurs when an MPLS-TE LSP traverses hops that are not in the LSP's headend's OSPF area and IS-IS level.

Interarea MPLS-TE allows you to configure an interarea traffic engineering (TE) label switched path (LSP) by specifying a loose source route of ABRs along the path. It is the then the responsibility of the ABR (having a complete view of both areas) to find a path obeying the TE LSP constraints within the next area to reach the next hop ABR (as specified on the headend). The same operation is performed by the last ABR connected to the tailend area to reach the tailend LSR.

You must be aware of these considerations when using loose hop optimization:

• You must specify the router ID of the ABR node (as opposed to a link address on the ABR).
• When multiarea is deployed in a network that contains subareas, you must enable MPLS-TE in the subarea for TE to find a path when loose hop is specified.
• You must specify the reachable explicit path for the interarea tunnel.

**Loose Hop Reoptimization**

Loose hop reoptimization allows the reoptimization of the tunnels spanning multiple areas and solves the problem which occurs when an MPLS-TE headend does not have visibility into other IGP areas.
Whenever the headend attempts to reoptimize a tunnel, it tries to find a better path to the ABR in the headend area. If a better path is found then the headend initiates the setup of a new LSP. In case a suitable path is not found in the headend area, the headend initiates a querying message. The purpose of this message is to query the ABRs in the areas other than the headend area to check if there exist any better paths in those areas. The purpose of this message is to query the ABRs in the areas other than the headend area, to check if a better path exists. If a better path does not exist, ABR forwards the query to the next router downstream. Alternatively, if better path is found, ABR responds with a special Path Error to the headend to indicate the existence of a better path outside the headend area. Upon receiving the Path Error that indicates the existence of a better path, the headend router initiates the reoptimization.

**ABR Node Protection**

Because one IGP area does not have visibility into another IGP area, it is not possible to assign backup to protect ABR node. To overcome this problem, node ID sub-object is added into the record route object of the primary tunnel so that at a PLR node, backup destination address can be checked against primary tunnel record-route object and assign a backup tunnel.

**Fast Reroute Node Protection**

If a link failure occurs within an area, the upstream router directly connected to the failed link generates an RSVP path error message to the headend. As a response to the message, the headend sends an RSVP path tear message and the corresponding path option is marked as invalid for a specified period and the next path-option (if any) is evaluated.

To retry the ABR immediately, a second path option (identical to the first one) should be configured. Alternatively, the retry period (path-option hold-down, 2 minutes by default) can be tuned to achieve a faster retry.

**Related Topics**
- Protecting MPLS Tunnels with Fast Reroute, page 38

**MPLS-TE Forwarding Adjacency**

The MPLS-TE Forwarding Adjacency feature allows a network administrator to handle a traffic engineering, label-switched path (LSP) tunnel as a link in an Interior Gateway Protocol (IGP) network based on the Shortest Path First (SPF) algorithm. A forwarding adjacency can be created between routers regardless of their location in the network.

**MPLS-TE Forwarding Adjacency Benefits**

TE tunnel interfaces are advertised in the IGP network just like any other links. Routers can then use these advertisements in their IGPs to compute the SPF even if they are not the head end of any TE tunnels.

**Related Topics**
- Configuring MPLS-TE Forwarding Adjacency, page 71
- Configure Forwarding Adjacency: Example, page 119

**MPLS-TE Forwarding Adjacency Restrictions**

The following restrictions are listed for the MPLS-TE Forwarding Adjacency feature:
Using the MPLS-TE Forwarding Adjacency feature increases the size of the IGP database by advertising a TE tunnel as a link.

The MPLS-TE Forwarding Adjacency feature is supported by Intermediate System-to-Intermediate System (IS-IS).

When the MPLS-TE Forwarding Adjacency feature is enabled on a TE tunnel, the link is advertised in the IGP network as a Type-Length-Value (TLV) 22 without any TE sub-TLV.

MPLS-TE forwarding adjacency tunnels must be configured bidirectionally.

**MPLS-TE Forwarding Adjacency Prerequisites**

Your network must support the following features before enabling the MPLS-TE Forwarding Adjacency feature:

- MPLS
- IP Cisco Express Forwarding
- Intermediate System-to-Intermediate System (IS-IS)

**Path Computation Element**

Path Computation Element (PCE) solves the specific issue of inter-domain path computation for MPLS-TE label switched path (LSPs), when the head-end router does not possess full network topology information (for example, when the head-end and tail-end routers of an LSP reside in different IGP areas).

PCE uses area border routers (ABRs) to compute a TE LSP spanning multiple IGP areas as well as computation of Inter-AS TE LSP.

PCE is usually used to define an overall architecture, which is made of several components, as follows:

**Path Computation Element (PCE)**

Represents a software module (which can be a component or application) that enables the router to compute paths applying a set of constraints between any pair of nodes within the router’s TE topology database. PCEs are discovered through IGP.

**Path Computation Client (PCC)**

Represents a software module running on a router that is capable of sending and receiving path computation requests and responses to and from PCEs. The PCC is typically an LSR (Label Switching Router).

**PCC-PCE communication protocol (PCEP)**

Specifies that PCEP is a TCP-based protocol defined by the IETF PCE WG, and defines a set of messages and objects used to manage PCEP sessions and to request and send paths for multi-domain TE LSPs. PCEP is used for communication between PCC and PCE (as well as between two PCEs) and employs IGP extensions to dynamically discover PCE.
Path computation elements provide support for the following message types and objects:

- Message types: Open, PCReq, PCRep, PCErr, Close
- Objects: OPEN, CLOSE, RP, END-POINT, LSPA, BANDWIDTH, METRIC, and NO-PATH

Related Topics
- Configuring a Path Computation Client, page 72
- Configuring a Path Computation Element Address, page 74
- Configuring PCE Parameters, page 75
- Configure PCE: Example, page 120

Path Protection

Path protection provides an end-to-end failure recovery mechanism (that is, full path protection) for MPLS-TE tunnels. A secondary Label Switched Path (LSP) is established, in advance, to provide failure protection for the protected LSP that is carrying a tunnel's TE traffic. When there is a failure on the protected LSP, the source router immediately enables the secondary LSP to temporarily carry the tunnel's traffic. If there is a failure on the secondary LSP, the tunnel no longer has path protection until the failure along the secondary path is cleared. Path protection can be used with a single area (OSPF or IS-IS), external BGP [eBGP], and static).

The failure detection mechanisms trigger a switchover to a secondary tunnel:

- Path error or resv-tear from Resource Reservation Protocol (RSVP) signaling
- Notification from the RSVP hello that a neighbor is lost
- Notification from the Bidirectional Forwarding Detection (BFD) protocol that a neighbor is lost
- Notification from the Interior Gateway Protocol (IGP) that the adjacency is down
• Local teardown of the protected tunnel's LSP due to preemption in order to signal higher priority LSPs, a Packet over SONET (POS) alarm, online insertion and removal (OIR), and so forth

An alternate recovery mechanism is Fast Reroute (FRR), which protects MPLS-TE LSPs only from link and node failures by locally repairing the LSPs at the point of failure.

Although not as fast as link or node protection, presignaling a secondary LSP is faster than configuring a secondary primary path option or allowing the tunnel's source router to dynamically recalculate a path. The actual recovery time is topology-dependent, and affected by delay factors such as propagation delay or switch fabric latency.

Related Topics

• Enabling Path Protection for an Interface, page 78
• Assigning a Dynamic Path Option to a Tunnel, page 80
• Forcing a Manual Switchover on a Path-Protected Tunnel, page 81
• Configuring the Delay the Tunnel Takes Before Reoptimization, page 82
• Configure Tunnels for Path Protection: Example, page 121

Prerequisites for Path Protection

• Ensure that your network supports MPLS-TE, Cisco Express Forwarding, and Intermediate System-to-Intermediate System (IS-IS) or Open Shortest Path First (OSPF).
• Enable MPLS.
• Configure TE on the routers.
• Configure a TE tunnel with a dynamic path option by using the path-option command with the dynamic keyword.

Restrictions for Path Protection

• Only Point-to-Point (P2P) tunnels are supported.
• Point-to-Multipoint (P2MP) TE tunnels are not supported.
• A maximum of one standby LSP is supported.
• There can be only one secondary path for each dynamic path option.
• Explicit path option can be configured for the path protected TE with the secondary path option as dynamic.
• Do not use link and node protection with path protection on the headend router.
• A maximum number of path protected tunnel TE heads is 2000.
• A maximum number of TE tunnel heads is equal to 4000.
• Path protection and FRR is mutually exclusive.
MPLS-TE Automatic Bandwidth

The MPLS-TE automatic bandwidth feature measures the traffic in a tunnel and periodically adjusts the signaled bandwidth for the tunnel.

These topics provide information about MPLS-TE automatic bandwidth:

MPLS-TE Automatic Bandwidth Overview

MPLS-TE automatic bandwidth is configured on individual Label Switched Paths (LSPs) at every head-end. MPLS-TE monitors the traffic rate on a tunnel interface. Periodically, MPLS-TE resizes the bandwidth on the tunnel interface to align it closely with the traffic in the tunnel. MPLS-TE automatic bandwidth can perform these functions:

- Monitors periodic polling of the tunnel output rate
- Resizes the tunnel bandwidth by adjusting the highest rate observed during a given period

For every traffic-engineered tunnel that is configured for an automatic bandwidth, the average output rate is sampled, based on various configurable parameters. Then, the tunnel bandwidth is readjusted automatically based upon either the largest average output rate that was noticed during a certain interval, or a configured maximum bandwidth value.

This table lists the automatic bandwidth functions.

Table 2: Automatic Bandwidth Variables

<table>
<thead>
<tr>
<th>Function</th>
<th>Command</th>
<th>Description</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application frequency</td>
<td>application</td>
<td>Configures how often the tunnel bandwidths changed for each tunnel. The application period is</td>
<td>24 hours</td>
</tr>
<tr>
<td></td>
<td>command</td>
<td>the period of A minutes between the bandwidth applications during which the output rate collection is done.</td>
<td></td>
</tr>
<tr>
<td>Requested bandwidth</td>
<td>bw-limit</td>
<td>Limits the range of bandwidth within the automatic-bandwidth feature that can request a bandwidth.</td>
<td>0 Kbps</td>
</tr>
<tr>
<td></td>
<td>command</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Collection frequency</td>
<td>auto-bw collect</td>
<td>Configures how often the tunnel output rate is polled globally for all tunnels.</td>
<td>5 min</td>
</tr>
<tr>
<td></td>
<td>command</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Highest collected</td>
<td>—</td>
<td>You cannot configure this value.</td>
<td>—</td>
</tr>
<tr>
<td>bandwidth</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Default Value

- You cannot configure this value.

## Command Function

<table>
<thead>
<tr>
<th>Function</th>
<th>Command</th>
<th>Description</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delta</td>
<td>—</td>
<td>You cannot configure this value.</td>
<td>—</td>
</tr>
</tbody>
</table>

The output rate on a tunnel is collected at regular intervals that are configured by using the **application** command in MPLS-TE auto bandwidth interface configuration mode. When the application period timer expires, and when the difference between the measured and the current bandwidth exceeds the adjustment threshold, the tunnel is reoptimized. Then, the bandwidth samples are cleared to record the new largest output rate at the next interval.

When reoptimizing the LSP with the new bandwidth, a new path request is generated. If the new bandwidth is not available, the last good LSP continues to be used. This way, the network experiences no traffic interruptions.

If minimum or maximum bandwidth values are configured for a tunnel, the bandwidth, which the automatic bandwidth signals, stays within these values.

---

**Note**

When more than 100 tunnels are **auto-bw** enabled, the algorithm will jitter the first application of every tunnel by a maximum of 20% (max 1 hour). The algorithm does this to avoid too many tunnels running auto bandwidth applications at the same time.

If a tunnel is shut down, and is later brought again, the adjusted bandwidth is lost and the tunnel is brought back with the initial configured bandwidth. In addition, the application period is reset when the tunnel is brought back.

### Related Topics

- Configuring the Collection Frequency, page 83
- Configuring the Automatic Bandwidth Functions, page 86
- Configure Automatic Bandwidth: Example, page 121

**Adjustment Threshold**

*Adjustment Threshold* is defined as a percentage of the current tunnel bandwidth and an absolute (minimum) bandwidth. Both thresholds must be fulfilled for the automatic bandwidth to resignal the tunnel. The tunnel bandwidth is resized only if the difference between the largest sample output rate and the current tunnel bandwidth is larger than the adjustment thresholds.

For example, assume that the automatic bandwidth is enabled on a tunnel in which the highest observed bandwidth B is 30 Mbps. Also, assume that the tunnel was initially configured for 45 Mbps. Therefore, the difference is 15 mbit/s. Now, assuming the default adjustment thresholds of 10% and 10kbps, the tunnel is signalled with 30 Mbps when the application timer expires. This is because 10% of 45 Mbit/s is 4.5 Mbit/s, which is smaller than 15 Mbit/s. The absolute threshold, which by default is 10 kbps, is also crossed.

**Overflow Detection**

Overflow detection is used if a bandwidth must be resized as soon as an overflow condition is detected, without having to wait for the expiry of an automatic bandwidth application frequency interval.

For overflow detection one configures a limit N, a percentage threshold Y% and optionally, a minimum bandwidth threshold Z. The percentage threshold is defined as the percentage of the actual signalled tunnel
bandwidth. When the difference between the measured bandwidth and the actual bandwidth are both larger than \( Y\% \) and \( Z \) threshold, for \( N \) consecutive times, then the system triggers an overflow detection.

The bandwidth adjustment by the overflow detection is triggered only by an increase of traffic volume through the tunnel, and not by a decrease in the traffic volume. When you trigger an overflow detection, the automatic bandwidth application interval is reset.

By default, the overflow detection is disabled and needs to be manually configured.

**Restrictions for MPLS-TE Automatic Bandwidth**

When the automatic bandwidth cannot update the tunnel bandwidth, the following restrictions are listed:

- Tunnel is in a fast reroute (FRR) backup, active, or path protect active state. This occurs because of the assumption that protection is a temporary state, and there is no need to reserve the bandwidth on a backup tunnel. You should prevent taking away the bandwidth from other primary or backup tunnels.

- Reoptimization fails to occur during a lockdown. In this case, the automatic bandwidth does not update the bandwidth unless the bandwidth application is manually triggered by using the `mpls traffic-eng auto-bw apply` command in EXEC mode.

**Point-to-Multipoint Traffic-Engineering**

**Point-to-Multipoint Traffic-Engineering Overview**

The Point-to-Multipoint (P2MP) Resource Reservation Protocol-Traffic Engineering (RSVP-TE) solution allows service providers to implement IP multicast applications, such as IPTV and real-time video, broadcast over the MPLS label switch network. The RSVP-TE protocol is extended to signal point-to-point (P2P) and P2MP label switched paths (LSPs) across the MPLS networks.

By using RSVP-TE extensions as defined in RFC 4875, multiple subLSPs are signaled for a given TE source. The P2MP tunnel is considered as a set of Source-to-Leaf (S2L) subLSPs that connect the TE source to multiple leaf Provider Edge (PE) nodes.

At the TE source, the ingress point of the P2MP-TE tunnel, IP multicast traffic is encapsulated with a unique MPLS label, which is associated with the P2MP-TE tunnel. The traffic continues to be label-switched in the P2MP tree. If needed, the labeled packet is replicated at branch nodes along the P2MP tree. When the labeled packet reaches the egress leaf (PE) node, the MPLS label is removed and forwarded onto the IP multicast tree across the PE-CE link.

To enable end-to-end IP multicast connectivity, RSVP is used in the MPLS-core for P2MP-TE signaling and PIM is used for PE-CE link signaling.

- All edge routers are running PIM-SSM or Source-Specific Multicast (SSM) to exchange multicast routing information with the directly-connected Customer Edge (CE) routers.

- In the MPLS network, RSVP P2MP-TE replaces PIM as the tree building mechanism, RSVP-TE grafts or prunes a given P2MP tree when the end-points are added or removed in the TE source configuration (explicit user operation).

These are the definitions for Point-to-Multipoint (P2MP) tunnels:

- **Source**: Configures the node in which Label Switched Path (LSP) signaling is initiated.
Point-to-Multipoint Traffic-Engineering Features

- P2MP RSVP-TE (RFC 4875) is supported. RFC 4875 is based on nonaggregate signaling; for example, per S2L signaling. Only P2MP LSP is supported.
- `interface tunnel-mte` command identifies the P2MP interface type.
- P2MP tunnel setup is supported with label replication.
- Fast-Reroute (FRR) link protection is supported with sub-50 msec for traffic loss.
- Explicit routing is supported by using under utilized links.
- Reoptimization is supported by calculating a better set of paths to the destination with no traffic loss.

**Note**

Per-S2L reoptimization is not supported.

- IPv4 and IPv6 payloads are supported.
- IPv4 and IPv6 multicast forwarding are supported on a P2MP tunnel interface through a static IGMP and MLD group configuration.
- Both IP multicast and P2MP Label Switch Multicast (LSM) coexist in the same network; therefore, both use the same forwarding plane (LFiB or MPLS Forwarding Infrastructure [MFI]).
- P2MP label replication supports only Source-Specific Multicast (SSM) traffic. SSM configuration supports the default value, none.
- Static mapping for multicast groups to the P2MP-TE tunnel is required.

Point-to-Multipoint Traffic-Engineering Benefits

- Single point of traffic control ensures that signaling and path engineering parameters (for example, protection and diversity) are configured only at the TE source node.
- Ability to configure explicit paths to enable optimized traffic distribution and prevention of single point of failures in the network.
- Link protection of MPLS-labeled traffic traversing branch paths of the P2MP-TE tree.
• Ability to do bandwidth Admission Control (AC) during setup and signaling of P2MP-TE paths in the MPLS network.

Related Topics
- Configure Point-to-Multipoint for the Source: Example, page 127
- Configure the Point-to-Multipoint Solution: Example, page 129
- Disable a Destination: Example, page 129
- Configure the Point-to-Multipoint Tunnel: Example, page 128
- Configure the Point-to-Multipoint Solution: Example, page 129
- Point-to-Multipoint RSVP-TE, page 24
- Path Option for Point-to-Multipoint RSVP-TE, page 25

Point-to-Multipoint RSVP-TE
RSVP-TE signals a P2MP tunnel base that is based on a manual configuration. If all Source-to-Leaf (S2L)s use an explicit path, the P2MP tunnel creates a static tree that follows a predefined path based on a constraint such as a deterministic Label Switched Path (LSP). If the S2L uses a dynamic path, RSVP-TE creates a P2MP tunnel base on the best path in the RSVP-TE topology. RSVP-TE supports bandwidth reservation for constraint-based routing.

RSVP-TE distributes stream information in which the topology tree does not change often (where the source and receivers are). For example, large scale video distribution between major sites is suitable for a subset of multicast applications. Because multicast traffic is already in the tunnel, the RSVP-TE tree is protected as long as you build a backup path.

Fast-Reroute (FRR) capability is supported for P2MP RSVP-TE by using the unicast link protection. You can choose the type of traffic to go to the backup link.

The P2MP tunnel is signaled by the dynamic and explicit path option in the IGP intra area. Only interArea and interAS, which are used for the P2MP tunnels, are signaled by the verbatim path option.

Related Topics
- Configure Point-to-Multipoint for the Source: Example, page 127
- Configure the Point-to-Multipoint Solution: Example, page 129
- Point-to-Multipoint Fast Reroute, page 24
- Path Option for Point-to-Multipoint RSVP-TE, page 25

Point-to-Multipoint Fast Reroute
MPLS-TE Fast Reroute (FRR) is a mechanism to minimize interruption in traffic delivery to a TE Label Switched Path (LSP) destination as a result of link or node failures. FRR enables temporarily fast switching of LSP traffic along an alternative backup path around a network failure, until the TE tunnel source signals a new end-to-end LSP.

The Point-of-Local Repair (PLR) is a node that selects a backup tunnel and switches the LSP traffic onto the backup tunnel in case a failure is detected. The receiver of the backup tunnel is referred to as the Merge Point (MP).

Both Point-to-Point (P2P) and P2MP-TE support only the Facility FRR method from RFC 4090.

Fast reroutable LSPs can coexist with fast reroutable P2P LSPs in a network. Node, link, and bandwidth protection for P2P LSPs are supported. Both MPLS-TE link and node protection rely on the fact that labels
for all primary LSPs and subLSPs are using the MPLS global label allocation. For example, one single (global)
label space is used for all MPLS-TE enabled physical interfaces on a given MPLS node.

Related Topics

- Point-to-Multipoint Traffic-Engineering Overview, page 22
- Point-to-Multipoint RSVP-TE, page 24

Point-to-Multipoint Label Switch Path

The Point-to-Multipoint Label Switch Path (P2M LSP) has only a single root, which is the Ingress Label
Switch Router (LSR). The P2M LSP is created based on a receiver that is connected to the Egress LSR. The
Egress LSR initiates the creation of the tree (for example, tunnel grafting or pruning is done by performing
an individual sub-LSP operation) by creating the Forwarding Equivalency Class (FEC) and Opaque Value.

Note

Grafting and pruning operate on a per destination basis.

The Opaque Value contains the stream information that uniquely identifies the tree to the root. To receive
label switched multicast packets, the Egress Provider Edge (PE) indicates to the upstream router (the next
hop closest to the root) which label it uses for the multicast source by applying the label mapping message.
The upstream router does not need to have any knowledge of the source; it needs only the received FEC to
identify the correct P2M LSP. If the upstream router does not have any FEC state, it creates it and installs
the assigned downstream outgoing label into the label forwarding table. If the upstream router is not the root
of the tree, it must forward the label mapping message to the next hop upstream. This process is repeated
hop-by-hop until the root is reached.

By using downstream allocation, the router that wants to receive the multicast traffic assigns the label for it.
The label request, which is sent to the upstream router, is similar to an unsolicited label mapping (that is, the
upstream does not request it). The upstream router that receives that label mapping uses the specific label to
send multicast packets downstream to the receiver. The advantage is that the router, which allocates the labels,
does not get into a situation where it has the same label for two different multicast sources. This is because it
manages its own label space allocation locally.

Path Option for Point-to-Multipoint RSVP-TE

P2MP tunnels are signaled by using the dynamic and explicit path-options in an IGP intra area. InterArea and
InterAS cases for P2MP tunnels are signaled by the verbatim path option.

Path options for P2MP tunnels are individually configured for each sub-LSP. Only one path option per sub-LSP
(destination) is allowed. You can choose whether the corresponding sub-LSP is dynamically or explicitly
routed. For the explicit option, you can configure the verbatim path option to bypass the topology database
lookup and verification for the specified destination.

Both dynamic and explicit path options are supported on a per destination basis by using the path-option
(P2MP-TE) command. In addition, you can combine both path options.

Explicit Path Option

Configures the intermediate hops that are traversed by a sub-LSP going from the TE source to the egress MPLS node. Although an explicit path configuration enables granular control sub-LSP paths in an MPLS network, multiple explicit paths are configured for specific network topologies with a limited number of (equal cost) links or paths.
Dynamic Path Option  Computes the IGP path of a P2MP tree sub-LSP that is based on the OSPF and ISIS algorithm. The TE source is dynamically calculated based on the IGP topology.

Dynamic Path Calculation Requirements  
Dynamic path calculation for each sub-LSP uses the same path parameters as those for the path calculation of regular point-to-point TE tunnels. As part of the sub-LSP path calculation, the link resource (bandwidth) is included, which is flooded throughout the MPLS network through the existing RSVP-TE extensions to OSPF and ISIS. Instead of dynamic calculated paths, explicit paths are also configured for one or more sub-LSPs that are associated with the P2MP-TE tunnel.

- OSPF or ISIS are used for each destination.
- TE topology and tunnel constraints are used to input the path calculation.
- Tunnel constraints such as affinity, bandwidth, and priorities are used for all destinations in a tunnel.
- Path calculation yields an explicit route to each destination.

Static Path Calculation Requirements  
The static path calculation does not require any new extensions to IGP to advertise link availability.

- Explicit path is required for every destination.
- Offline path calculation is used.
- TE topology database is not needed.
- If the topology changes, reoptimization is not required.

Related Topics  
- Configure the Point-to-Multipoint Tunnel: Example, page 128
- Configure the Point-to-Multipoint Solution: Example, page 129
- Point-to-Multipoint Traffic-Engineering Overview, page 22
- Point-to-Multipoint RSVP-TE, page 24

MPLS Traffic Engineering Shared Risk Link Groups  
Shared Risk Link Groups (SRLG) in MPLS traffic engineering refer to situations in which links in a network share a common fiber (or a common physical attribute). These links have a shared risk, and that is when one link fails, other links in the group might fail too.

OSPF and Intermediate System-to-Intermediate System (IS-IS) flood the SRLG value information (including other TE link attributes such as bandwidth availability and affinity) using a sub-type length value (sub-TLV), so that all routers in the network have the SRLG information for each link.

To activate the SRLG feature, configure the SRLG value of each link that has a shared risk with another link. A maximum of 30 SRLGs per interface is allowed. You can configure this feature on multiple interfaces including the bundle interface.
The Explicit Path configuration allows you to configure the explicit path. An IP explicit path is a list of IP addresses, each representing a node or link in the explicit path.

The MPLS Traffic Engineering (TE)—IP Explicit Address Exclusion feature provides a means to exclude a link or node from the path for an Multiprotocol Label Switching (MPLS) TE label-switched path (LSP).

This feature is enabled through the `explicit-path` command that allows you to create an IP explicit path and enter a configuration submode for specifying the path. The feature adds to the submode commands of the `exclude-address` command for specifying addresses to exclude from the path.

The feature also adds to the submode commands of the `exclude-srlg` command that allows you to specify the IP address to get SRLGs to be excluded from the explicit path.

If the excluded address or excluded srlg for an MPLS TE LSP identifies a flooded link, the constraint-based shortest path first (CSPF) routing algorithm does not consider that link when computing paths for the LSP. If the excluded address specifies a flooded MPLS TE router ID, the CSPF routing algorithm does not allow paths for the LSP to traverse the node identified by the router ID.

**Fast ReRoute with SRLG Constraints**

Fast ReRoute (FRR) protects MPLS TE Label Switch Paths (LSPs) from link and node failures by locally repairing the LSPs at the point of failure. This protection allows data to continue to flow on LSPs, 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.

Backup tunnels that bypass only a single link of the LSP's path provide Link Protection. They protect LSPs by specifying the protected link IP addresses to extract SRLG values that are to be excluded from the explicit path, thereby 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. Figure 7: NHOP Backup Tunnel with SRLG constraint, page 28 illustrates an NHOP backup tunnel.

**Figure 7: NHOP Backup Tunnel with SRLG constraint**

In the topology shown in the above figure, the backup tunnel path computation can be performed in this manner:

- Get all SRLG values from the exclude-SRLG link (SRLG values 5 and 6)
- Mark all the links with the same SRLG value to be excluded from SPF
- Path computation as CSPF R2->R6->R7->R3

FRR provides Node Protection for LSPs. Backup tunnels that bypass next-hop nodes along LSP paths are called **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 when a node along their path fails, by enabling the node upstream to the point of failure to reroute the LSPs and their traffic, around the failed node to the next-next hop. They also protect LSPs by specifying the protected link IP addresses that are to be excluded from the explicit path, and the SRLG values associated with the IP addresses excluded from the explicit path.
NNHOP backup tunnels also provide protection from link failures by bypassing the failed link as well as the node. Figure 8: NNHOP Backup Tunnel with SRLG constraint, page 29 illustrates an NNHOP backup tunnel.

**Figure 8: NNHOP Backup Tunnel with SRLG constraint**

In the topology shown in the above figure, the backup tunnel path computation can be performed in this manner:

- Get all SRLG values from the exclude-SRLG link (SRLG values 5 and 6)
- Mark all links with the same SRLG value to be excluded from SPF
- Verify path with SRLG constraint
- Path computation as CSPF R2->R9->R10->R4

**Importance of Protection**

This section describes the following:

- Delivery of Packets During a Failure
- Multiple Backup Tunnels Protecting the Same Interface

**Delivery of Packets During a Failure**

Backup tunnels that terminate at the NNHOP protect both the downstream link and node. This provides protection for link and node failures.

**Multiple Backup Tunnels Protecting the Same Interface**

- 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 falls 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).

**SRLG Limitations**

There are few limitations to the configured SRLG feature:

- The `exclude-address` and `exclude-srlg` options are not allowed in the IP `explicit path strict-address` network.
- Whenever SRLG values are modified after tunnels are signalled, they are verified dynamically in the next path verification cycle.

**How to Implement Traffic Engineering**

Traffic engineering requires coordination among several global neighbor routers, creating traffic engineering tunnels, setting up forwarding across traffic engineering tunnels, setting up FRR, and creating differential service.

These procedures are used to implement MPLS-TE:

**Building MPLS-TE Topology**

Perform this task to configure MPLS-TE topology (required for traffic engineering tunnel operations).

**Before You Begin**

Before you start to build the MPLS-TE topology, you must have enabled:

- IGP such as OSPF or IS-IS for MPLS-TE.
- MPLS Label Distribution Protocol (LDP).
- RSVP on the port interface.
- Stable router ID is required at either end of the link to ensure that the link is successful. If you do not assign a router ID, the system defaults to the global router ID. Default router IDs are subject to change, which can result in an unstable link.
- If you are going to use nondefault holdtime or intervals, you must decide the values to which they are set.
SUMMARY STEPS

1. configure
2. mpls traffic-eng
3. interface type interface-path-id
4. exit
5. exit
6. router ospf process-name
7. area area-id
8. exit
9. mpls traffic-eng router-id type interface-path-id
10. Use one of the following commands:
    • end
    • commit
11. (Optional) show mpls traffic-eng topology
12. (Optional) show mpls traffic-eng link-management advertisements

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>configure</td>
<td>Enters the configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>mpls traffic-eng</td>
<td>Enters MPLS-TE configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config)# mpls traffic-eng</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td></td>
</tr>
<tr>
<td>interface type interface-path-id</td>
<td>Enables traffic engineering on a particular interface on the originating node and enters MPLS-TE interface configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)# interface POS0/6/0/0</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mpls-te-if)#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td></td>
</tr>
<tr>
<td>exit</td>
<td>Exits the current configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mpls-te-if)# exit</td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>------------------</td>
<td>---------</td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)#</td>
<td></td>
</tr>
</tbody>
</table>

**Step 5**

**exit**

Example:

RP/0/RSP0/CPU0:router(config-mpls-te)# exit
RP/0/RSP0/CPU0:router(config)#

Exits the current configuration mode.

**Step 6**

**router ospf process-name**

Example:

RP/0/RSP0/CPU0:router(config)# router ospf 1

Enters a name for the OSPF process.

**Step 7**

**area area-id**

Example:

RP/0/RSP0/CPU0:router(config-router)# area 0

Configures an area for the OSPF process.

- Backbone areas have an area ID of 0.
- Non-backbone areas have a non-zero area ID.

**Step 8**

**exit**

Example:

RP/0/RSP0/CPU0:router(config-ospf-ar)# exit
RP/0/RSP0/CPU0:router(config-ospf)#

Exits the current configuration mode.

**Step 9**

**mpls traffic-eng router-id type interface-path-id**

Example:

RP/0/RSP0/CPU0:router(config-ospf)# mpls traffic-eng router-id Loopback0

Sets the MPLS-TE loopback interface.

**Step 10**

Use one of the following commands:

- **end**
- **commit**

Example:

RP/0/RSP0/CPU0:router(config-ospf)# end
or
RP/0/RSP0/CPU0:router(config-ospf)# commit

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

  Uncommitted changes found, commit them before exiting(yes/no/cancel)?
  (cancel):

  - Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
  - Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
Creating an MPLS-TE Tunnel

Creating an MPLS-TE tunnel is a process of customizing the traffic engineering to fit your network topology. Perform this task to create an MPLS-TE tunnel after you have built the traffic engineering topology.

Before You Begin

The following prerequisites are required to create an MPLS-TE tunnel:

- You must have a router ID for the neighboring router.
- Stable router ID is required at either end of the link to ensure that the link is successful. If you do not assign a router ID to the routers, the system defaults to the global router ID. Default router IDs are subject to change, which can result in an unstable link.
- If you are going to use nondefault holdtime or intervals, you must decide the values to which they are set.

Related Topics

- How MPLS-TE Works, page 3
- Build MPLS-TE Topology and Tunnels: Example, page 115
### SUMMARY STEPS

1. configure
2. interface tunnel-te *tunnel-id*
3. destination *ip-address*
4. ipv4 unnumbered *type interface-path-id*
5. path-option *preference - priority dynamic*
6. signalled- bandwidth \{bandwidth [class-type \textit{ct}] | sub-pool bandwidth\}
7. Use one of the following commands:
   - end
   - commit

8. (Optional) show mpls traffic-eng tunnels
9. (Optional) show ipv4 interface brief
10. (Optional) show mpls traffic-eng link-management admission-control

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> interface tunnel-te <em>tunnel-id</em></td>
<td>Configures an MPLS-TE tunnel interface.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router# interface tunnel-te 1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> destination <em>ip-address</em></td>
<td>Assigns a destination address on the new tunnel.</td>
</tr>
<tr>
<td>Example:</td>
<td>The destination address is the remote node’s MPLS-TE router ID.</td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-if)# destination 192.168.92.125</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> ipv4 unnumbered <em>type interface-path-id</em></td>
<td>Assigns a source address so that forwarding can be performed on the new tunnel. Loopback is commonly used as the interface type.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-if)# ipv4 unnumbered Loopback0</td>
<td></td>
</tr>
</tbody>
</table>

---

---
<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>path-option preference - priority dynamic</td>
<td>Sets the path option to dynamic and assigns the path ID.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> RP/0/RSP0/CPU0:router(config-if)# path-option 1 dynamic</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>signalled-bandwidth {bandwidth [class-type ct]</td>
<td>sub-pool bandwidth}</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> RP/0/RSP0/CPU0:router(config-if)# signalled-bandwidth 100</td>
<td></td>
</tr>
</tbody>
</table>
| 7    | Use one of the following commands:  

  - end  
  - commit | Saves configuration changes.  

  - When you issue the **end** command, the system prompts you to commit changes:  

    Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:  

    - Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.  
    - Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.  
    - Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.  

    - Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session. |
|      | **Example:** RP/0/RSP0/CPU0:router(config-if)# end or  

  or  

  RP/0/RSP0/CPU0:router(config-if)# commit | |
| 8    | show mpls traffic-eng tunnels | (Optional)  

  Verifies that the tunnel is connected (in the UP state) and displays all configured TE tunnels. |
|      | **Example:** RP/0/RSP0/CPU0:router# show mpls traffic-eng tunnels | |
| 9    | show ipv4 interface brief | (Optional)  

  Displays all TE tunnel interfaces. |
|      | **Example:** RP/0/RSP0/CPU0:router# show ipv4 interface | |
### Configuring Forwarding over the MPLS-TE Tunnel

Perform this task to configure forwarding over the MPLS-TE tunnel created in the previous task. This task allows MPLS packets to be forwarded on the link between network neighbors.

#### Before You Begin

The following prerequisites are required to configure forwarding over the MPLS-TE tunnel:

- You must have a router ID for the neighboring router.
- Stable router ID is required at either end of the link to ensure that the link is successful. If you do not assign a router ID to the routers, the system defaults to the global router ID. Default router IDs are subject to change, which can result in an unstable link.

#### SUMMARY STEPS

1. `configure`
2. `interface tunnel-te tunnel-id`
3. `ipv4 unnumbered type interface-path-id`
4. `autoroute announce`
5. `exit`
6. `router static address-family ipv4 unicast prefix mask ip-address interface type`
7. Use one of the following commands:
   
   - `end`
   - `commit`
8. (Optional) `ping {ip-address | hostname}`
9. (Optional) `show mpls traffic-eng autoroute`

---

**Related Topics**

- How MPLS-TE Works, page 3
- Build MPLS-TE Topology and Tunnels: Example, page 115
- Building MPLS-TE Topology, page 30
### DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>configure</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>interface tunnel-te tunnel-id</td>
<td>Enters MPLS-TE interface configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RSP0/CPU0:router(config)# interface tunnel-te 1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>ipv4 unnumbered type interface-path-id</td>
<td>Assigns a source address so that forwarding can be performed on the new tunnel.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RSP0/CPU0:router(config-if)# ipv4 unnumbered Loopback0</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>autoroute announce</td>
<td>Enables messages that notify the neighbor nodes about the routes that are forwarding.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RSP0/CPU0:router(config-if)# autoroute announce</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>exit</td>
<td>Exits the current configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RSP0/CPU0:router(config-if)# exit</td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td>router static address-family ipv4 unicast prefix mask ip-address interface type</td>
<td>Enables a route using IP version 4 addressing, identifies the destination address and the tunnel where forwarding is enabled. This configuration is used for static routes when the autoroute announce command is not used.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RSP0/CPU0:router(config)# router static address-family ipv4 unicast 2.2.2.2/32 tunnel-te 1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 7</strong></td>
<td>Use one of the following commands:</td>
<td>Saves configuration changes.</td>
</tr>
<tr>
<td></td>
<td>• end</td>
<td>• When you issue the end command, the system prompts you to commit changes:</td>
</tr>
<tr>
<td></td>
<td>• commit</td>
<td>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</td>
</tr>
</tbody>
</table>
### Command or Action

**Example:**

RP/0/RSP0/CPU0:router(config)# end
or
RP/0/RSP0/CPU0:router(config)# commit

**Purpose**

- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
- Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
- Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

### Step 8

**ping {ip-address | hostname}**

**Example:**

RP/0/RSP0/CPU0:router# ping 192.168.12.52

(Optional) Checks for connectivity to a particular IP address or host name.

### Step 9

**show mpls traffic-eng autoroute**

**Example:**

RP/0/RSP0/CPU0:router# show mpls traffic-eng autoroute

(Optional) Verifies forwarding by displaying what is advertised to IGP for the TE tunnel.

---

**Related Topics**

- Overview of MPLS Traffic Engineering, page 2
- Creating an MPLS-TE Tunnel, page 33

---

**Protecting MPLS Tunnels with Fast Reroute**

Perform this task to protect MPLS-TE tunnels, as created in the previous task.

---

**Note**

Although this task is similar to the previous task, its importance makes it necessary to present as part of the tasks required for traffic engineering on Cisco IOS XR software.

---

**Before You Begin**

The following prerequisites are required to protect MPLS-TE tunnels:

- You must have a router ID for the neighboring router.
- Stable router ID is required at either end of the link to ensure that the link is successful. If you do not assign a router ID to the routers, the system defaults to the global router ID. Default router IDs are subject to change, which can result in an unstable link.
- You must first configure a primary tunnel.

**SUMMARY STEPS**

1. `configure`
2. `interface tunnel-te tunnel-id`
3. `fast-reroute`
4. `exit`
5. `mpls traffic-eng`
6. `interface type interface-path-id`
7. `backup-path tunnel-te tunnel-number`
8. `exit`
9. `exit`
10. `interface tunnel-te tunnel-id`
11. `backup-bw {backup bandwidth | sub-pool {bandwidth | unlimited} | global-pool {bandwidth | unlimited}}`
12. `ipv4 unnumbered type interface-path-id`
13. `path-option preference-priority {explicit name explicit-path-name}`
14. `destination ip-address`
15. Use one of the following commands:
   - `end`
   - `commit`

16. (Optional) `show mpls traffic-eng tunnels backup`
17. (Optional) `show mpls traffic-eng tunnels protection frr`
18. (Optional) `show mpls traffic-eng fast-reroute database`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Step 1</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>configure</code></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example: <code>RP/0/RSP0/CPU0:router# configure</code></td>
<td></td>
</tr>
</tbody>
</table>
### Purpose

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>interface tunnel-te tunnel-id</strong></td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router# interface tunnel-te 1</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td><strong>fast-reroute</strong></td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router(config-if)# fast-reroute</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td><strong>exit</strong></td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router(config-if)# exit</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td><strong>mpls traffic-eng</strong></td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router(config)# mpls traffic-eng</td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td><strong>interface type interface-path-id</strong></td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router(config-mpls-te-if)# interface pos0/6/0/0</td>
</tr>
<tr>
<td><strong>Step 7</strong></td>
<td><strong>backup-path tunnel-te tunnel-number</strong></td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router(config-mpls-te-if)# backup-path tunnel-te 2</td>
</tr>
<tr>
<td><strong>Step 8</strong></td>
<td><strong>exit</strong></td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router(config-mpls-te-if)# exit</td>
</tr>
</tbody>
</table>

**Protecting MPLS Tunnels with Fast Reroute**

- Configures an MPLS-TE tunnel interface.
- Enables fast reroute.
- Exits the current configuration mode.
- Enters MPLS-TE configuration mode.
- Enables traffic engineering on a particular interface on the originating node.
- Sets the backup path to the backup tunnel.
- Exits the current configuration mode.
<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td><code>exit</code></td>
<td>Exits the current configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>exit</code></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>interface <code>tunnel-te tunnel-id</code></td>
<td>Configures an MPLS-TE tunnel interface.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>interface tunnel-te 2</code></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>backup-bw `{backup bandwidth</td>
<td>sub-pool `{bandwidth</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>backup-bw</code> <code>global-pool 5000</code></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>ipv4 unnumbered <code>type interface-path-id</code></td>
<td>Assigns a source address to set up forwarding on the new tunnel.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>ipv4 unnumbered Loopback0</code></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>path-option <code>preference-priority {explicit name explicit-path-name}</code></td>
<td>Sets the path option to explicit with a given name (previously configured) and assigns the path ID.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>path-option explicit name backup-path</code></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>destination <code>ip-address</code></td>
<td>Assigns a destination address on the new tunnel.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>destination 192.168.92.125</code></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>Use one of the following commands:</td>
<td>Saves configuration changes.</td>
</tr>
<tr>
<td></td>
<td>Note</td>
<td>Because the default tunnel priority is 7, tunnels use the default TE class map.</td>
</tr>
<tr>
<td></td>
<td>Note</td>
<td>When you configure TE tunnel with multiple protection on its path and merge point is the same node for more than one protection, you must configure record-route for that tunnel.</td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
<td></td>
</tr>
</tbody>
</table>
| • end  
  • commit | • When you issue the **end** command, the system prompts you to commit changes:  
  Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:  
  - Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.  
  - Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.  
  - Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.  
  • Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session. |

**Step 16**  
**show mpls traffic-eng tunnels backup**  
*(Optional)*  
Displays the backup tunnel information.  
*Example:*  
```
RP/0/RSP0/CPU0:router# show mpls traffic-eng tunnels backup
```

**Step 17**  
**show mpls traffic-eng tunnels protection frr**  
*(Optional)*  
Displays the tunnel protection information for Fast-Reroute (FRR).  
*Example:*  
```
RP/0/RSP0/CPU0:router# show mpls traffic-eng tunnels protection frr
```

**Step 18**  
**show mpls traffic-eng fast-reroute database**  
*(Optional)*  
Displays the protected tunnel state (for example, the tunnel’s current ready or active state).  
*Example:*  
```
RP/0/RSP0/CPU0:router# show mpls traffic-eng fast-reroute database
```

**Related Topics**  
- Fast Reroute, page 11  
- Fast Reroute Node Protection, page 16  
- Creating an MPLS-TE Tunnel, page 33  
- Configuring Forwarding over the MPLS-TE Tunnel, page 36
Enabling an AutoTunnel Backup

Perform this task to configure the AutoTunnel Backup feature. By default, this feature is disabled. You can configure the AutoTunnel Backup feature for each interface. It has to be explicitly enabled for each interface or link.

**SUMMARY STEPS**

1. configure
2. ipv4 unnumbered mpls traffic-eng Loopback 0
3. mpls traffic-eng
4. auto-tunnel backup timers removal unused frequency
5. auto-tunnel backup tunnel-id min min max
6. Use one of the following commands:
   - end
   - commit
7. show mpls traffic-eng auto-tunnel backup summary

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router# configure</td>
</tr>
</tbody>
</table>
| **Step 2** ipv4 unnumbered mpls traffic-eng Loopback 0 | Configures the globally configured IPv4 address that can be used by the AutoTunnel Backup Tunnels.  
*Note* Loopback 0 is the router ID. The AutoTunnel Backup tunnels will not come up until a global IPv4 address is configured. |
| Example:          | RP/0/RSP0/CPU0:router(config)#ipv4 unnumbered mpls traffic-eng Loopback 0 |
| **Step 3** mpls traffic-eng | Enters MPLS-TE configuration mode. |
| Example:          | RP/0/RSP0/CPU0:router(config)# mpls traffic-eng |
| **Step 4** auto-tunnel backup timers removal unused frequency | Configures how frequently a timer scans the backup automatic tunnels and removes tunnels that are not in use.  
*Note* You can also configure the auto-tunnel backup command at mpls traffic-eng interface mode. |
| Example:          | RP/0/RSP0/CPU0:router(config-mpls-te)# auto-tunnel backup timers removal unused 20  
*Note* Use the frequency argument to scan the backup automatic tunnel. Range is 0 to 10080. |
Removing an AutoTunnel Backup

To remove all the backup autotunnels, perform this task to remove the AutoTunnel Backup feature.
SUMMARY STEPS

1. clear mpls traffic-eng auto-tunnel backup unused { all | tunnel-te number}
2. Use one of the following commands:
   - end
   - commit
3. show mpls traffic-eng auto-tunnel summary

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> clear mpls traffic-eng auto-tunnel backup unused { all</td>
<td>tunnel-te number}</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router# clear mpls traffic-eng auto-tunnel backup unused all</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> Use one of the following commands:</td>
<td>Saves configuration changes.</td>
</tr>
<tr>
<td>• end</td>
<td></td>
</tr>
<tr>
<td>• commit</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config)# end or</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config)# commit</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> show mpls traffic-eng auto-tunnel summary</td>
<td>Displays information about MPLS-TE autotunnels including the ones removed.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router# show mpls traffic-eng auto-tunnel summary</td>
<td></td>
</tr>
</tbody>
</table>
Establishing MPLS Backup AutoTunnels to Protect Fast Reroutable TE LSPs

To establish an MPLS backup autotunnel to protect fast reroutable TE LSPs, perform these steps:

**SUMMARY STEPS**

1. `configure`
2. `mpls traffic-eng`
3. `interface type interface-path-id`
4. `auto-tunnel backup`
5. Use one of the following commands:
   - `end`
   - `commit`
6. `show mpls traffic-eng auto-tunnel backup summary`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td><code>configure</code></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td><code>mpls traffic-eng</code></td>
<td>Enters MPLS-TE configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RSP0/CPU0:router(config)# mpls traffic-eng</td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td><code>interface type interface-path-id</code></td>
<td>Enables traffic engineering on a specific interface on the originating node.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)# interface POS 0/6/0/0</td>
<td></td>
</tr>
<tr>
<td>Step 4</td>
<td><code>auto-tunnel backup</code></td>
<td>Enables an autotunnel backup feature for the specified interface.</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>You cannot configure the static backup on the similar link.</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RSP0/CPU0:router(config-mpls-te-if)# auto-tunnel backup</td>
<td></td>
</tr>
<tr>
<td>Step 5</td>
<td>Use one of the following commands:</td>
<td>Saves configuration changes.</td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
<td></td>
</tr>
<tr>
<td>• end</td>
<td>When you issue the <strong>end</strong> command, the system prompts you to commit changes:</td>
<td></td>
</tr>
<tr>
<td>• commit</td>
<td>Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Entering <strong>yes</strong> saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Entering <strong>no</strong> exits the configuration session and returns the router to EXEC mode without committing the configuration changes.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Entering <strong>cancel</strong> leaves the router in the current configuration session without exiting or committing the configuration changes.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Use the <strong>commit</strong> command to save the configuration changes to the running configuration file and remain within the configuration session.</td>
<td></td>
</tr>
</tbody>
</table>

**Example:**
```
RP/0/RSP0/CPU0:router(config)# end

RP/0/RSP0/CPU0:router(config)# commit
```

**Step 6**

<table>
<thead>
<tr>
<th>show mpls traffic-eng auto-tunnel backup summary</th>
<th>Displays information about configured MPLS-TE backup autotunnels.</th>
</tr>
</thead>
</table>

**Example:**
```
RP/0/RSP0/CPU0:router# show mpls traffic-eng auto-tunnel backup summary
```

**Related Topics**

- Backup AutoTunnels, page 4
- Configure the MPLS-TE Auto-Tunnel Backup: Example, page 123

**Establishing Next-Hop Tunnels with Link Protection**

To establish a next-hop tunnel and link protection on the primary tunnel, perform these steps:
### SUMMARY STEPS

1. `configure`
2. `mpls traffic-eng`
3. `interface type interface-path-id`
4. `auto-tunnel backup nhop-only`
5. `auto-tunnel backup exclude srlg` [preferred]
6. Use one of the following commands:
   - `end`
   - `commit`
7. `show mpls traffic-eng tunnels number detail`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><code>configure</code></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td><code>mpls traffic-eng</code></td>
<td>Enters MPLS-TE configuration mode.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config)# mpls traffic-eng</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td><code>interface type interface-path-id</code></td>
<td>Enables traffic engineering on a specific interface on the originating node.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)# interface POS 0/6/0/0</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td><code>auto-tunnel backup nhop-only</code></td>
<td>Enables the creation of dynamic NHOP backup tunnels. By default, both NHOP and NNHOP protection are enabled. <strong>Note</strong> Using this nhop-only option, only link protection is provided.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-mpls-te-if)# auto-tunnel backup nhop-only</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td><code>auto-tunnel backup exclude srlg</code> [preferred]</td>
<td>Enables the exclusion of SRLG values on a given link for the AutoTunnel backup associated with a given interface. The preferred option allows the AutoTunnel Backup tunnels to come up even if no path excluding all SRLG is found.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-mpls-te-if)# auto-tunnel backup exclude srlg preferred</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Use one of the following commands:</td>
<td>Saves configuration changes.</td>
</tr>
</tbody>
</table>

---

Implementing MPLS Traffic Engineering

Establishing Next-Hop Tunnels with Link Protection
Configuring a Prestandard DS-TE Tunnel

Perform this task to configure a Prestandard DS-TE tunnel.

Before You Begin

The following prerequisites are required to configure a Prestandard DS-TE tunnel:

• You must have a router ID for the neighboring router.

• Stable router ID is required at either end of the link to ensure that the link is successful. If you do not assign a router ID to the routers, the system defaults to the global router ID. Default router IDs are subject to change, which can result in an unstable link.
SUMMARY STEPS

1. configure
2. rsvp interface type interface-path-id
3. bandwidth [total reservable bandwidth] [bc0 bandwidth] [global-pool bandwidth] [sub-pool reservable-bw]
4. exit
5. exit
6. interface tunnel-te tunnel-id
7. signalled-bandwidth {bandwidth [class-type ct] | sub-pool bandwidth}
8. Use one of the following commands:
   • end
   • commit

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>configure</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>rsvp interface type interface-path-id</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config)# rsvp interface pos0/6/0/0</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>bandwidth [total reservable bandwidth] [bc0 bandwidth] [global-pool bandwidth] [sub-pool reservable-bw]</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-rsvp-if)# bandwidth 100 150 sub-pool 50</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>exit</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-rsvp-if)# exit</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-rsvp)#</td>
</tr>
</tbody>
</table>
### Purpose

**Command or Action**

**Purpose**

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>exit</td>
<td>Exits the current configuration mode.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-rsvp)# exit</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config)# exit</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>interface tunnel-te tunnel-id</td>
<td>Configures an MPLS-TE tunnel interface.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config)# interface tunnel-te 2</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>signalled-bandwidth {bandwidth [class-type ct]</td>
<td>Sets the bandwidth required on this interface. Because the default tunnel priority is 7, tunnels use the default TE class map (namely, class-type 1, priority 7).</td>
</tr>
<tr>
<td></td>
<td>sub-pool bandwidth}</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-if)# signalled-bandwidth sub-pool 10</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Use one of the following commands:</td>
<td>Saves configuration changes.</td>
</tr>
<tr>
<td></td>
<td>• end</td>
<td>• When you issue the end command, the system prompts you to commit changes:</td>
</tr>
<tr>
<td></td>
<td>• commit</td>
<td>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-if)# end or</td>
<td>• Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-if)# commit</td>
<td>• Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.</td>
</tr>
</tbody>
</table>

### Related Topics

- Prestandard DS-TE Mode, page 8
- Configure IETF DS-TE Tunnels: Example, page 116
Configuring an IETF DS-TE Tunnel Using RDM

Perform this task to create an IETF mode DS-TE tunnel using RDM.

Before You Begin

The following prerequisites are required to create an IETF mode DS-TE tunnel using RDM:

- You must have a router ID for the neighboring router.
- Stable router ID is required at either end of the link to ensure that the link is successful. If you do not assign a router ID to the routers, the system defaults to the global router ID. Default router IDs are subject to change, which can result in an unstable link.

SUMMARY STEPS

1. `configure`
2. `rsvp interface type interface-path-id`
3. `bandwidth rdm {total-reservable-bw | bc0 | global-pool} {sub-pool | bc1 reservable-bw}`
4. `exit`
5. `exit`
6. `mpls traffic-eng`
7. `ds-te mode ietf`
8. `exit`
9. `interface tunnel-te tunnel-id`
10. `signalled-bandwidth {bandwidth [class-type ct] | sub-pool bandwidth}`
11. Use one of the following commands:
    - `end`
    - `commit`

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> <code>configure</code></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Step 2</strong> <code>rsvp interface type interface-path-id</code></td>
<td>Enters RSVP configuration mode and selects an RSVP interface.</td>
</tr>
</tbody>
</table>

Example:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# rsvp interface
```
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>pos0/6/0/0</td>
<td>Sets the reserved RSVP bandwidth available on this interface by using the Russian Doll Model (RDM) bandwidth constraints model. The range for the <code>total reserve bandwidth</code> argument is 0 to 4294967295. <strong>Note</strong>: Physical interface bandwidth is not used by MPLS-TE.</td>
</tr>
</tbody>
</table>

**Step 3**

`bandwidth rdm {total-reservable-bw | bc0 | global-pool} {sub-pool | bc1 reservable-bw}`

**Example:**
```
RP/0/RSP0/CPU0:router(config-rsvp-if)# bandwidth rdm 100 150
```

**Step 4**

`exit`

**Example:**
```
RP/0/RSP0/CPU0:router(config-rsvp-if)# exit
RP/0/RSP0/CPU0:router(config-rsvp)
```

**Step 5**

`exit`

**Example:**
```
RP/0/RSP0/CPU0:router(config-rsvp)# exit
RP/0/RSP0/CPU0:router(config)
```

**Step 6**

`mpls traffic-eng`

**Example:**
```
RP/0/RSP0/CPU0:router(config)# mpls traffic-eng
RP/0/RSP0/CPU0:router(config-mpls-te)#
```

**Step 7**

`ds-te mode ietf`

**Example:**
```
RP/0/RSP0/CPU0:router(config-mpls-te)# ds-te mode ietf
```

**Step 8**

`exit`

**Example:**
```
RP/0/RSP0/CPU0:router(config-mpls-te)# exit
```

**Step 9**

`interface tunnel-te tunnel-id`

**Example:**
```
RP/0/RSP0/CPU0:router(config)# interface tunnel-te 4
```
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>RP/0/RSP0/CPU0:router(config-if)#</td>
<td></td>
</tr>
</tbody>
</table>

**Step 10**

`signalled-bandwidth {bandwidth [class-type ct] | sub-pool bandwidth}

Example:

RP/0/RSP0/CPU0:router(config-if)#
signalled-bandwidth 10 class-type 1

**Step 11**

Use one of the following commands:

- `end`
- `commit`

Example:

RP/0/RSP0/CPU0:router(config-if)# end
RP/0/RSP0/CPU0:router(config-if)# commit

Saves configuration changes.

- When you issue the `end` command, the system prompts you to commit changes:

  Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:

  - Entering `yes` saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
  - Entering `no` exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
  - Entering `cancel` leaves the router in the current configuration session without exiting or committing the configuration changes.

- Use the `commit` command to save the configuration changes to the running configuration file and remain within the configuration session.

**Related Topics**

- Russian Doll Bandwidth Constraint Model, page 9

**Configuring an IETF DS-TE Tunnel Using MAM**

Perform this task to configure an IETF mode differentiated services traffic engineering tunnel using the Maximum Allocation Model (MAM) bandwidth constraint model.

**Before You Begin**

The following prerequisites are required to configure an IETF mode differentiated services traffic engineering tunnel using the MAM bandwidth constraint model:

- You must have a router ID for the neighboring router.
• Stable router ID is required at either end of the link to ensure that the link is successful. If you do not assign a router ID to the routers, the system defaults to the global router ID. Default router IDs are subject to change, which can result in an unstable link.

SUMMARY STEPS

1. configure
2. rsvp interface type interface-path-id
3. bandwidth mam {total reservable bandwidth | max-reservable-bw maximum-reservable-bw} [bc0 reservable bandwidth] [bc1 reservable bandwidth]
4. exit
5. exit
6. mpls traffic-eng
7. ds-te mode ietf
8. ds-te bc-model mam
9. exit
10. interface tunnel-te tunnel-id
11. signalled-bandwidth {bandwidth [class-type ct] | sub-pool bandwidth}
12. Use one of the following commands:
   • end
   • commit

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>configure</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Enters RSVP configuration mode and selects the RSVP interface.</td>
</tr>
<tr>
<td>rsvp interface type interface-path-id</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config)# rsvp interface pos0/6/0/0</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Sets the reserved RSVP bandwidth available on this interface.</td>
</tr>
<tr>
<td>bandwidth mam {total reservable bandwidth</td>
<td>max-reservable-bw maximum-reservable-bw} [bc0 reservable bandwidth] [bc1 reservable bandwidth]</td>
</tr>
<tr>
<td>Example:</td>
<td>Note Physical interface bandwidth is not used by MPLS-TE.</td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-rsvp-if)# bandwidth</td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>------------------</td>
<td>---------</td>
</tr>
<tr>
<td><code>mam max-reservable-bw 400 bc0 300 bc1 200</code></td>
<td></td>
</tr>
</tbody>
</table>

### Step 4

`exit`  

*Example:*  
`RP/0/RSP0/CPU0:router(config-rsvp-if)# exit`  
`RP/0/RSP0/CPU0:router(config-rsvp)#`

### Step 5

`exit`  

*Example:*  
`RP/0/RSP0/CPU0:router(config-rsvp)# exit`  
`RP/0/RSP0/CPU0:router(config)#`

### Step 6

`mpls traffic-eng`  

*Example:*  
`RP/0/RSP0/CPU0:router(config)# mpls traffic-eng`  
`RP/0/RSP0/CPU0:router(config-mpls-te)#`

### Step 7

`ds-te modeietf`  

*Example:*  
`RP/0/RSP0/CPU0:router(config-mpls-te)# ds-te mode ietf`

### Step 8

`ds-te bc-model mam`  

*Example:*  
`RP/0/RSP0/CPU0:router(config-mpls-te)# ds-te bc-model mam`

### Step 9

`exit`  

*Example:*  
`RP/0/RSP0/CPU0:router(config-mpls-te)# exit`

### Step 10

`interface tunnel-te tunnel-id`  

*Example:*  
`RP/0/RSP0/CPU0:router(config)# interface tunnel-te 4`  
`RP/0/RSP0/CPU0:router(config-if)#`
### Purpose

**Command or Action**

**Step 11**  
configured-bandwidth \( \{ \text{bandwidth [class-type ct]} \mid \text{sub-pool bandwidth} \} \)

Example:

```
RP/0/RSP0/CPU0:router(config-rsvp-if)# configured-bandwidth 10 class-type 1
```

**Step 12**  
Use one of the following commands:

- `end`
- `commit`

Example:

```
RP/0/RSP0/CPU0:router(config-rsvp-if)# end
```

Uncommitted changes found, commit them before exiting(yes/no/cancel)?

- [cancel]:
  - Entering `yes` saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
  - Entering `no` exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
  - Entering `cancel` leaves the router in the current configuration session without exiting or committing the configuration changes.

- Use the `commit` command to save the configuration changes to the running configuration file and remain within the configuration session.

### Related Topics

- Maximum Allocation Bandwidth Constraint Model, page 9

### Configuring MPLS -TE and Fast-Reroute on OSPF

Perform this task to configure MPLS-TE and Fast Reroute (FRR) on OSPF.
Before You Begin

Note
Only point-to-point (P2P) interfaces are supported for OSPF multiple adjacencies. These may be either native P2P interfaces or broadcast interfaces on which the OSPF P2P configuration command is applied to force them to behave as P2P interfaces as far as OSPF is concerned. This restriction does not apply to IS-IS.

The tunnel-te interface is not supported under IS-IS.

SUMMARY STEPS

1. `configure`
2. `interface tunnel-te tunnel-id`
3. `path-option [protecting ] preference-priority {dynamic [pce [address ipv4 address] | explicit {name pathname | identifier path-number } ] [isis instance name {level level} ] [ospf instance name {area area ID} ] } [verbatim] [lockdown]`
4. Repeat Step 3 as many times as needed.
5. Use one of the following commands:
   - `end`
   - `commit`
6. `show mpls traffic-eng tunnels [tunnel-number]`

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><code>configure</code></td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><code>interface tunnel-te tunnel-id</code></td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>`path-option [protecting ] preference-priority {dynamic [pce [address ipv4 address]</td>
</tr>
</tbody>
</table>
### Configuring the Ignore Integrated IS-IS Overload Bit Setting in MPLS-TE

Perform this task to configure an overload node avoidance in MPLS-TE. When the overload bit is enabled, tunnels are brought down when the overload node is found in the tunnel path.

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-if)# path-option 1 explicit identifier 6 ospf green area 0</td>
<td>Configures another explicit path option.</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td></td>
</tr>
<tr>
<td>Repeat Step 3 as many times as needed.</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-if)# path-option 2 explicit name 234 ospf 3 area 7 verbatim</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td></td>
</tr>
<tr>
<td>Use one of the following commands:</td>
<td></td>
</tr>
<tr>
<td>• end</td>
<td>Saves configuration changes.</td>
</tr>
<tr>
<td>• commit</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-if)# end</td>
<td></td>
</tr>
<tr>
<td>or</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-if)# commit</td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td></td>
</tr>
<tr>
<td>show mpls traffic-eng tunnels [tunnel-number]</td>
<td>Displays information about MPLS-TE tunnels.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router# show mpls traffic-eng tunnels 1</td>
<td></td>
</tr>
</tbody>
</table>
SUMMARY STEPS

1. configure
2. mpls traffic-eng
3. path-selection ignore overload {head | mid | tail}
4. Use one of these commands:
   • end
   • commit

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>configure</td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router# configure</td>
</tr>
</tbody>
</table>

| **Step 2**        | Enters MPLS-TE configuration mode. |
| mpls traffic-eng  | Example: |
|                   | RP/0/RSP0/CPU0:router(config)# mpls traffic-eng |
|                   | RP/0/RSP0/CPU0:router(config-mpls-te)# |

| **Step 3**        | Ignores the Intermediate System-to-Intermediate System (IS-IS) overload bit setting for MPLS-TE. |
|                   | If set-overload-bit is set by IS-IS on the head router, the tunnels stay up. |
| path-selection ignore overload {head | mid | tail} | Example: |
|                   | RP/0/RSP0/CPU0:router(config-mpls-te)# path-selection ignore overload head |

<p>| <strong>Step 4</strong>        | Saves configuration changes. |
|                   | • When you issue the end command, the system prompts you to commit changes: |
|                   | Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: |
|                   | • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. |
|                   | • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. |
| Use one of these commands: | Example: |
|                   | RP/0/RSP0/CPU0:router(config-mpls-te)# end |
|                   | or |
|                   | RP/0/RSP0/CPU0:router(config-mpls-te)# |</p>
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| commit            | • Entering `cancel` leaves the router in the current configuration session without exiting or committing the configuration changes.  
• Use the `commit` command to save the configuration changes to the running configuration file and remain within the configuration session. |

**Related Topics**
- Ignore Intermediate System-to-Intermediate System Overload Bit Setting in MPLS-TE, page 12
- Configure the Ignore IS-IS Overload Bit Setting in MPLS-TE: Example, page 117

## Configuring Flexible Name-based Tunnel Constraints

To fully configure MPLS-TE flexible name-based tunnel constraints, you must complete these high-level tasks in order:

1. Assigning Color Names to Numeric Values, page 61
2. Associating Affinity-Names with TE Links, page 63
3. Associating Affinity Constraints for TE Tunnels, page 64

### Assigning Color Names to Numeric Values

The first task in enabling the new coloring scheme is to assign a numerical value (in hexadecimal) to each value (color).

**Note**

An affinity color name cannot exceed 64 characters. An affinity value cannot exceed a single digit. For example, magenta1.

**SUMMARY STEPS**

1. configure
2. mpls traffic-eng
3. affinity-map affinity name  {affinity value | bit-position value}
4. Use one of the following commands:
   • end
   • commit
### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><code>configure</code></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><code>configure</code></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router# <code>configure</code></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Enters MPLS-TE configuration mode.</td>
</tr>
<tr>
<td><code>mpls traffic-eng</code></td>
<td>Enters MPLS-TE configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Enters MPLS-TE configuration mode.</td>
</tr>
<tr>
<td><code>mpls traffic-eng</code></td>
<td>Enters MPLS-TE configuration mode.</td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config)# <code>mpls traffic-eng</code></td>
<td>Enters MPLS-TE configuration mode.</td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)#</td>
<td>Enters MPLS-TE configuration mode.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Enters an affinity name and a map value by using a color name (repeat this command to assign multiple colors up to a maximum of 64 colors). An affinity color name cannot exceed 64 characters. The value you assign to a color name must be a single digit.</td>
</tr>
<tr>
<td>`affinity-map affinity name {affinity value</td>
<td>bit-position value}`</td>
</tr>
<tr>
<td>Example:</td>
<td>Enters an affinity name and a map value by using a color name (repeat this command to assign multiple colors up to a maximum of 64 colors). An affinity color name cannot exceed 64 characters. The value you assign to a color name must be a single digit.</td>
</tr>
<tr>
<td><code>affinity-map red 1</code></td>
<td>Enters an affinity name and a map value by using a color name (repeat this command to assign multiple colors up to a maximum of 64 colors). An affinity color name cannot exceed 64 characters. The value you assign to a color name must be a single digit.</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>Saves configuration changes.</td>
</tr>
<tr>
<td>Use one of the following commands:</td>
<td>Saves configuration changes.</td>
</tr>
<tr>
<td>• <code>end</code></td>
<td>• When you issue the <code>end</code> command, the system prompts you to commit changes:</td>
</tr>
<tr>
<td>• <code>commit</code></td>
<td>Uncommitted changes found, commit them before exiting(yes/no/cancel)?</td>
</tr>
<tr>
<td>Example:</td>
<td>[cancel]:</td>
</tr>
<tr>
<td><code>end</code> or</td>
<td>• Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)# <code>end</code> or</td>
<td>• Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</td>
</tr>
<tr>
<td><code>commit</code></td>
<td>• Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.</td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)# <code>commit</code></td>
<td>• Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.</td>
</tr>
</tbody>
</table>

**Related Topics**

- Flexible Name-based Tunnel Constraints, page 13
Associating Affinity-Names with TE Links

The next step in the configuration of MPLS-TE Flexible Name-based Tunnel Constraints is to assign affinity names and values to TE links. You can assign up to a maximum of 32 colors. Before you assign a color to a link, you must define the name-to-value mapping for each color.

**SUMMARY STEPS**

1. configure
2. mpls traffic-eng
3. interface type interface-path-id
4. attribute-names attribute name
5. Use one of the following commands:
   - end
   - commit

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>configure</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td>mpls traffic-eng</td>
<td>Enters MPLS-TE configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config)# mpls traffic-eng</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)#</td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td>interface type interface-path-id</td>
<td>Enables MPLS-TE on an interface and enters MPLS-TE interface configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)# interface tunnel-te 2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-mpls-te-if)#</td>
<td></td>
</tr>
<tr>
<td>Step 4</td>
<td>attribute-names attribute name</td>
<td>Assigns colors to TE links over the selected interface.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-mpls-te-if)#</td>
<td></td>
</tr>
</tbody>
</table>
### Command or Action

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>attribute-names red</td>
<td>Saves configuration changes.</td>
</tr>
</tbody>
</table>

#### Step 5

Use one of the following commands:

- end
- commit

**Example:**

```
RP/0/RSP0/CPU0:router(config-mpls-te-if)#
end
```

or

```
RP/0/RSP0/CPU0:router(config-mpls-te-if)#
commit
```

- When you issue the `end` command, the system prompts you to commit changes:
  ```
  Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
  ```
  - Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
  - Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
  - Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.

- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

---

**Related Topics**

- Flexible Name-based Tunnel Constraints, page 13
- Configure Flexible Name-based Tunnel Constraints: Example, page 117
- Assigning Color Names to Numeric Values, page 61

---

**Associating Affinity Constraints for TE Tunnels**

The final step in the configuration of MPLS-TE Flexible Name-based Tunnel Constraints requires that you associate a tunnel with affinity constraints.

Using this model, there are no masks. Instead, there is support for four types of affinity constraints:

- include
- include-strict
- exclude
- exclude-all

**Note**

For the affinity constraints above, all but the exclude-all constraint may be associated with up to 10 colors.
SUMMARY STEPS

1. configure
2. interface tunnel-te tunnel-id
3. affinity \{affinity-value mask mask-value \| exclude name \| exclude -all \| include name \| include-strict name\}
4. Use one of the following commands:
   - end
   - commit

DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>configure</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td>interface tunnel-te tunnel-id</td>
<td>Configures an MPLS-TE tunnel interface.</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router(config)# interface tunnel-te 1</td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td>affinity {affinity-value mask mask-value | exclude name | exclude -all | include name | include-strict name}</td>
<td>Configures link attributes for links comprising a tunnel. You can have up to ten colors. Multiple include statements can be specified under tunnel configuration. With this configuration, a link is eligible for CSPF if it has at least a red color or has at least a green color. Thus, a link with red and any other colors as well as a link with green and any additional colors meet the above constraint.</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router(config-if)# affinity include red</td>
<td></td>
</tr>
<tr>
<td>Step 4</td>
<td>Use one of the following commands: • end • commit</td>
<td>Saves configuration changes. • When you issue the end command, the system prompts you to commit changes: Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router(config-if)# end or RP/0/RSP0/CPU0:router(config-if)# commit</td>
<td></td>
</tr>
</tbody>
</table>
### Configuring IS-IS to Flood MPLS-TE Link Information

Perform this task to configure a router running the Intermediate System-to-Intermediate System (IS-IS) protocol to flood MPLS-TE link information into multiple IS-IS levels. This procedure shows how to enable MPLS-TE in both IS-IS Level 1 and Level 2.

#### SUMMARY STEPS

1. configure
2. router isis instance-id
3. net network-entity-title
4. address-family {ipv4 | ipv6} {unicast}
5. metric-style wide
6. mpls traffic-eng level
7. Use one of the following commands:
   - end
   - commit

#### DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>configure</td>
<td>Enters global configuration mode.</td>
</tr>
</tbody>
</table>

Example:

```
RP/0/RSP0/CPU0:router# configure
```
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td><code>router isis instance-id</code></td>
<td>Enters an IS-IS instance.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>RP/0/RSP0/CPU0:router(config)# router isis 1</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td></td>
</tr>
<tr>
<td><code>net network-entity-title</code></td>
<td>Enters an IS-IS network entity title (NET) for the routing process.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>RP/0/RSP0/CPU0:router(config-isis)# net 47.0001.0000.0000.0002.00</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td></td>
</tr>
<tr>
<td>`address-family {ipv4</td>
<td>ipv6} {unicast}`</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>RP/0/RSP0/CPU0:router(config-isis)# address-family ipv4 unicast</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td></td>
</tr>
<tr>
<td><code>metric-style wide</code></td>
<td>Enters the new-style type, length, and value (TLV) objects.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>RP/0/RSP0/CPU0:router(config-isis-af)# metric-style wide</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td></td>
</tr>
<tr>
<td><code>mpls traffic-eng level</code></td>
<td>Enters the required MPLS-TE level or levels.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>RP/0/RSP0/CPU0:router(config-isis-af)# mpls traffic-eng level-1-2</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 7</strong></td>
<td></td>
</tr>
<tr>
<td>Use one of the following commands:</td>
<td>Saves configuration changes.</td>
</tr>
<tr>
<td>• <code>end</code></td>
<td>• When you issue the <code>end</code> command, the system prompts you to commit changes:</td>
</tr>
<tr>
<td>• <code>commit</code></td>
<td>Uncommitted changes found, commit them before exiting(yes/no/cancel)?</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>[cancel]:</td>
</tr>
<tr>
<td><code>RP/0/RSP0/CPU0:router(config-isis-af)# end</code> or <code>commit</code></td>
<td>• Entering <code>yes</code> saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>• Entering <code>no</code> exits the configuration session and returns the router to EXEC mode without committing the configuration changes.</td>
</tr>
</tbody>
</table>
Configuring an OSPF Area of MPLS-TE

Perform this task to configure an OSPF area for MPLS-TE in both the OSPF backbone area 0 and area 1.

SUMMARY STEPS

1. configure
2. router ospf process-name
3. mpls traffic-eng router-id type interface-path-id
4. area area-id
5. interface type interface-path-id
6. Use one of the following commands:
   - end
   - commit

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>configure</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>router ospf process-name</td>
<td>Enters a name that uniquely identifies an OSPF routing process.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config)# router ospf 100</td>
<td>process-name</td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>------------------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td><strong>mpls traffic-eng router-id type interface-path-id</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><strong>RP/0/RSP0/CPU0:router(config-ospf)# mpls traffic-eng router-id Loopback0</strong></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td><strong>area area-id</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><strong>RP/0/RSP0/CPU0:router(config-ospf)# area 0</strong></td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td><strong>interface type interface-path-id</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><strong>RP/0/RSP0/CPU0:router(config-ospf-ar)# interface POS 0/2/0/0</strong></td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td>Use one of the following commands:</td>
</tr>
<tr>
<td></td>
<td>• <strong>end</strong></td>
</tr>
</tbody>
</table>
| | • **commit** | Uncommitted changes found, commit them before exiting(yes/no/cancel)?
| | **Example:** | {cancel}: |
| | **RP/0/RSP0/CPU0:router(config-ospf-ar)# end** | • Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. |
| | **or** | • Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes. |
| | **RP/0/RSP0/CPU0:router(config-ospf-ar)# commit** | • Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes. |
| | | • Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session. |

### Configuring Explicit Paths with ABRs Configured as Loose Addresses

Perform this task to specify an IPv4 explicit path with ABRs configured as loose addresses.
SUMMARY STEPS

1. configure
2. explicit-path name name
3. index index-id next-address [loose] ipv4 unicast ip-address
4. Use one of the following commands:
   - end
   - commit

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router# configure</td>
</tr>
<tr>
<td><strong>Step 2</strong> explicit-path name name</td>
<td>Enters a name for the explicit path.</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router(config)# explicit-path name interarea1</td>
</tr>
<tr>
<td><strong>Step 3</strong> index index-id next-address [loose] ipv4 unicast ip-address</td>
<td>Includes an address in an IP explicit path of a tunnel.</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router(config-expl-path)# index 1 next-address loose ipv4 unicast 10.10.10.10</td>
</tr>
<tr>
<td><strong>Step 4</strong> Use one of the following commands:</td>
<td>Saves configuration changes.</td>
</tr>
<tr>
<td>• end</td>
<td>• When you issue the end command, the system prompts you to commit changes:</td>
</tr>
<tr>
<td>• commit</td>
<td>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</td>
</tr>
<tr>
<td>Example:</td>
<td>• Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-expl-path)# end</td>
<td>• Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.</td>
</tr>
<tr>
<td>or</td>
<td>RP/0/RSP0/CPU0:router(config-expl-path)# commit</td>
</tr>
</tbody>
</table>
Purpose

Command or Action | Purpose
--- | ---

- Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.

- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring MPLS-TE Forwarding Adjacency

Perform this task to configure forwarding adjacency on a specific tunnel-te interface.

### SUMMARY STEPS

1. **configure**
2. **interface tunnel-te tunnel-id**
3. **forwarding-adjacency holdtime** *value*
4. Use one of the following commands:
   - **end**
   - **commit**

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td><strong>configure</strong></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router# <strong>configure</strong></td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td><strong>interface tunnel-te tunnel-id</strong></td>
<td>Enters MPLS-TE interface configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router(config)# <strong>interface tunnel-te 1</strong></td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td><strong>forwarding-adjacency holdtime</strong> <em>value</em></td>
<td>Configures forwarding adjacency using an optional specific holdtime value. By default, this value is 0 (milliseconds).</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router(config-if)#</td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
<td></td>
</tr>
<tr>
<td>-------------------------</td>
<td>-------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>forwarding-adjacency holdtime 60</td>
<td>Saves configuration changes.</td>
<td></td>
</tr>
</tbody>
</table>

**Step 4**

Use one of the following commands:
- **end**
- **commit**

**Example:**

```
RP/0/RSP0/CPU0:router(config-if)# end
```

```
RP/0/RSP0/CPU0:router(config-if)# commit
```

- When you issue the **end** command, the system prompts you to commit changes:

```
Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:
```

  - Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
  - Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
  - Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.

- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

**Related Topics**

- MPLS-TE Forwarding Adjacency Benefits, page 16
- Configure Forwarding Adjacency: Example, page 119

**Configuring a Path Computation Client and Element**

Perform these tasks to configure Path Computation Client (PCC) and Path Computation Element (PCE):

- Configuring a Path Computation Client, page 72
- Configuring a Path Computation Element Address, page 74
- Configuring PCE Parameters, page 75

**Configuring a Path Computation Client**

Perform this task to configure a TE tunnel as a PCC.

**Note**

Only one TE-enabled IGP instance can be used at a time.
SUMMARY STEPS

1. configure
2. interface tunnel-te tunnel-id
3. path-option preference-priority dynamic pce
4. Use one of the following commands:
   - end
   - commit

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>configure</td>
<td>Example: <code>RP/0/RSP0/CPU0:router# configure</code></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Enters MPLS-TE interface configuration mode and enables traffic engineering on a particular interface on the originating node.</td>
</tr>
<tr>
<td>interface tunnel-te tunnel-id</td>
<td>Example: <code>RP/0/RSP0/CPU0:router(config)# interface tunnel-te 6</code></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Configures a TE tunnel as a PCC.</td>
</tr>
<tr>
<td>path-option preference-priority dynamic pce</td>
<td>Example: <code>RP/0/RSP0/CPU0:router(config-if)# path-option 1 dynamic pce</code></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>Saves configuration changes.</td>
</tr>
<tr>
<td>Use one of the following commands:</td>
<td>- When you issue the <code>end</code> command, the system prompts you to commit changes:</td>
</tr>
<tr>
<td>- end</td>
<td>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</td>
</tr>
<tr>
<td>- commit</td>
<td>- Entering <code>yes</code> saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</td>
</tr>
<tr>
<td>Example: <code>RP/0/RSP0/CPU0:router(config-if)# end</code> or <code>RP/0/RSP0/CPU0:router(config-if)# commit</code></td>
<td>- Entering <code>no</code> exits the configuration session and returns the router to EXEC mode without committing the configuration changes.</td>
</tr>
</tbody>
</table>
### Configuring a Path Computation Client and Element Address

Perform this task to configure a PCE address.

**Note** Only one TE-enabled IGP instance can be used at a time.

#### SUMMARY STEPS

1. `configure`
2. `mpls traffic-eng`
3. `pce address ipv4 address`
4. Use one of the following commands:
   - `end`
   - `commit`

#### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> <code>configure</code></td>
<td>Enters global configuration mode.</td>
</tr>
</tbody>
</table>

Example:
```
RP/0/RSP0/CPU0:router# configure
```
### Implementing MPLS Traffic Engineering

#### Configuring a Path Computation Client and Element

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong> mpls traffic-eng</td>
<td>Enters the MPLS-TE configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config)# mpls traffic-eng</td>
<td></td>
</tr>
</tbody>
</table>

| **Step 3** pce address ipv4 address | Configures a PCE IPv4 address. |
| Example:                          |                               |
| RP/0/RSP0/CPU0:router(config-mpls-te)# pce address ipv4 10.1.1.1 |                               |

| **Step 4** Use one of the following commands: | Saves configuration changes. |
|       • end | When you issue the **end** command, the system prompts you to commit changes: |
|       • commit | Uncommitted changes found, commit them before exiting(yes/no/cancel)? |
| Example: |                                           |
| RP/0/RSP0/CPU0:router(config-mpls-te)# end | (cancel): |
| or | • Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. |
| RP/0/RSP0/CPU0:router(config-mpls-te)# commit | • Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes. |
| | • Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes. |

- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

**Related Topics**

- Path Computation Element, page 17
- Configure PCE: Example, page 120

**Configuring PCE Parameters**

Perform this task to configure PCE parameters, including a static PCE peer, periodic reoptimization timer values, and request timeout values.
SUMMARY STEPS

1. configure
2. mpls traffic-eng
3. pce address ipv4 address
4. pce peer ipv4 address
5. pce keepalive interval
6. pce deadtimer value
7. pce reoptimize value
8. pce request-timeout value
9. pce tolerance keepalive value
10. Use one of the following commands:
    • end
    • commit
11. show mpls traffic-eng pce peer [address | all]
12. show mpls traffic-eng pce tunnels

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example: RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> mpls traffic-eng</td>
<td>Enters MPLS-TE configuration mode.</td>
</tr>
<tr>
<td>Example: RP/0/RSP0/CPU0:router(config)# mpls traffic-eng</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> pce address ipv4 address</td>
<td>Configures a PCE IPv4 address.</td>
</tr>
<tr>
<td>Example: RP/0/RSP0/CPU0:router(config-mpls-te)# pce address ipv4 10.1.1.1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> pce peer ipv4 address</td>
<td>Configures a static PCE peer address. PCE peers are also discovered dynamically through OSPF or ISIS.</td>
</tr>
<tr>
<td>Example: RP/0/RSP0/CPU0:router(config-mpls-te)# pce</td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>peer address ipv4 10.1.1.1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> pce keepalive interval</td>
<td>Configures a PCEP keepalive interval. The range is from 0 to 255 seconds. When the keepalive interval is 0, the LSR does not send keepalive messages.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)# pce keepalive 10</td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong> pce deadtimer value</td>
<td>Configures a PCE deadtimer value. The range is from 0 to 255 seconds. When the dead interval is 0, the LSR does not timeout a PCEP session to a remote peer.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)# pce deadtimer 50</td>
<td></td>
</tr>
<tr>
<td><strong>Step 7</strong> pce reoptimize value</td>
<td>Configures a periodic reoptimization timer value. The range is from 60 to 604800 seconds. When the dead interval is 0, the LSR does not timeout a PCEP session to a remote peer.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)# pce reoptimize 200</td>
<td></td>
</tr>
<tr>
<td><strong>Step 8</strong> pce request-timeout value</td>
<td>Configures a PCE request-timeout. Range is from 5 to 100 seconds. PCC or PCE keeps a pending path request only for the request-timeout period.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)# pce request-timeout 10</td>
<td></td>
</tr>
<tr>
<td><strong>Step 9</strong> pce tolerance keepalive value</td>
<td>Configures a PCE tolerance keepalive value (which is the minimum acceptable peer proposed keepalive).</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)# pce tolerance keepalive 10</td>
<td></td>
</tr>
<tr>
<td><strong>Step 10</strong> Use one of the following commands:</td>
<td>Saves configuration changes.</td>
</tr>
<tr>
<td>• end</td>
<td></td>
</tr>
<tr>
<td>• commit</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)# end or</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)# commit</td>
<td></td>
</tr>
</tbody>
</table>
# Configuring Path Protection on MPLS-TE

These tasks show how to configure path protection on MPLS-TE:

## Enabling Path Protection for an Interface

Perform this task to enable path protection for a given tunnel interface.

### SUMMARY STEPS

1. configure
2. interface tunnel-te tunnel-id
3. path-protection
4. Use one of the following commands:
   - end
   - commit
5. show mpls traffic-eng tunnels [tunnel-number]
### DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td><strong>configure</strong></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td><strong>interface tunnel-te tunnel-id</strong></td>
<td>Configures an MPLS-TE tunnel interface and enables traffic engineering on a particular interface on the originating node.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RSP0/CPU0:router(config)# interface tunnel-te 6</td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td><strong>path-protection</strong></td>
<td>Enables path protection on the tunnel-te interface.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RSP0/CPU0:router(config-if)# path-protection</td>
<td></td>
</tr>
<tr>
<td>Step 4</td>
<td>Use one of the following commands:</td>
<td>Saves configuration changes.</td>
</tr>
<tr>
<td>• <strong>end</strong></td>
<td></td>
<td>- When you issue the <strong>end</strong> command, the system prompts you to commit changes:</td>
</tr>
<tr>
<td>• <strong>commit</strong></td>
<td></td>
<td>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RSP0/CPU0:router(config-if)# end</td>
<td></td>
</tr>
<tr>
<td></td>
<td>or</td>
<td>○ Entering <strong>yes</strong> saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-if)# commit</td>
<td>○ Entering <strong>no</strong> exits the configuration session and returns the router to EXEC mode without committing the configuration changes.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>○ Entering <strong>cancel</strong> leaves the router in the current configuration session without exiting or committing the configuration changes.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Use the <strong>commit</strong> command to save the configuration changes to the running configuration file and remain within the configuration session.</td>
</tr>
<tr>
<td>Step 5</td>
<td><strong>show mpls traffic-eng tunnels [tunnel-number]</strong></td>
<td>Displays information that path protection is enabled on the tunnel-te interface for tunnel number 6.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RSP0/CPU0:router# show mpls</td>
<td></td>
</tr>
</tbody>
</table>
Assigning a Dynamic Path Option to a Tunnel

Perform this task to assign a secondary path option in case there is a link or node failure along a path and all interfaces in your network are not protected.

SUMMARY STEPS

1. configure
2. interface tunnel-te tunnel-id
3. path-option preference-priority dynamic
4. Use one of the following commands:
   • end
   • commit
5. show mpls traffic-eng tunnels [tunnel-number]

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 configure</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example: RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td>Step 2 interface tunnel-te tunnel-id</td>
<td>Configures an MPLS-TE tunnel interface and enables traffic engineering on a particular interface on the originating node.</td>
</tr>
<tr>
<td>Example: RP/0/RSP0/CPU0:router(config)# interface tunnel-te 6</td>
<td></td>
</tr>
<tr>
<td>Step 3 path-option preference-priority dynamic</td>
<td>Configures a secondary path option for an MPLS-TE tunnel.</td>
</tr>
<tr>
<td>Example: RP/0/RSP0/CPU0:router(config-if)#</td>
<td></td>
</tr>
</tbody>
</table>
### Step 4

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>path-option 10 dynamic</td>
<td>Saves configuration changes.</td>
</tr>
</tbody>
</table>

Use one of the following commands:

- **end**
- **commit**

**Example:**

```
RP/0/RSP0/CPU0:router(config-if)# end
```

---

### Step 5

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show mpls traffic-eng tunnels [tunnel-number]</td>
<td>Displays information about the secondary path option that on the tunnel-te interface for tunnel number 6.</td>
</tr>
</tbody>
</table>

**Example:**

```
RP/0/RSP0/CPU0:router# show mpls traffic-eng tunnels 6
```

---

**Related Topics**

- Path Protection, page 18
- Configure Tunnels for Path Protection: Example, page 121

---

**Forcing a Manual Switchover on a Path-Protected Tunnel**

Perform this task to force a manual switchover on a path-protected tunnel.

**SUMMARY STEPS**

1. `mpls traffic-eng path-protection switchover tunnel-te tunnel-ID`
**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Forces the path protection switchover of the Point-to-Point (P2P) tunnel on the tunnel-te interface.</td>
</tr>
<tr>
<td><strong>mpls traffic-eng path-protection switchover tunnel-te tunnel-ID</strong></td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
</tbody>
</table>

**Related Topics**
- Path Protection, page 18
- Configure Tunnels for Path Protection: Example, page 121

**Configuring the Delay the Tunnel Takes Before Reoptimization**

Perform this task to configure the time between when a path-protection switchover event is effected on a tunnel head to when a reoptimization is performed on that tunnel. This timer affects only the required reoptimization that is attempted due to a switchover and does not override the global reoptimization timer.

**SUMMARY STEPS**

1. configure
2. mpls traffic-eng
3. reoptimize timers delay path-protection seconds
4. Use one of the following commands:
   - end
   - commit

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>configure</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router# configure</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Enters MPLS-TE configuration mode.</td>
</tr>
<tr>
<td>mpls traffic-eng</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router# mpls traffic-eng</td>
</tr>
</tbody>
</table>
### Configuring the Automatic Bandwidth

Perform these tasks to configure the automatic bandwidth:

#### Configuring the Collection Frequency

Perform this task to configure the collection frequency. You can configure only one global collection frequency.

---

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 3** reoptimize timers delay path-protection seconds | Adjusts the number of seconds that the tunnel takes before triggering reoptimization after switchover has happened.  
**Note** The restriction is that at least one dynamic path-option must be configured for a standby LSP to come up. The strict (explicit) path option is not supported for the standby LSP. |
| **Example:** RP/0/RSP0/CPU0:router(config-mpls-te)# reoptimize timers delay path-protection 180 |  |
| **Step 4** Use one of the following commands:  
• end  
• commit | Saves configuration changes.  
• When you issue the `end` command, the system prompts you to commit changes:  
Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:  
• Entering `yes` saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.  
• Entering `no` exits the configuration session and returns the router to EXEC mode without committing the configuration changes.  
• Entering `cancel` leaves the router in the current configuration session without exiting or committing the configuration changes.  
• Use the `commit` command to save the configuration changes to the running configuration file and remain within the configuration session. |
| **Example:**  
RP/0/RSP0/CPU0:router(config-mpls-te)# end  
or  
RP/0/RSP0/CPU0:router(config-mpls-te)# commit |  |
SUMMARY STEPS

1. configure
2. mpls traffic-eng
3. auto-bw collect frequency minutes
4. Use one of the following commands:
   • end
   • commit
5. show mpls traffic-eng tunnels [auto-bw]

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example: RP/O/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> mpls traffic-eng</td>
<td>Enters MPLS-TE configuration mode.</td>
</tr>
<tr>
<td>Example: RP/O/RSP0/CPU0:router(config)# mpls traffic-eng RP/O/RSP0/CPU0:router(config-mpls-te)#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> auto-bw collect frequency minutes</td>
<td>Configures the automatic bandwidth collection frequency, and controls the manner in which the bandwidth for a tunnel collects output rate information; but does not adjust the tunnel bandwidth.</td>
</tr>
<tr>
<td>Example: RP/O/RSP0/CPU0:router(config-mpls-te)# auto-bw collect frequency 1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> Use one of the following commands:</td>
<td>Saves configuration changes.</td>
</tr>
<tr>
<td>• end</td>
<td>• When you issue the end command, the system prompts you to commit changes:</td>
</tr>
<tr>
<td>• commit</td>
<td>Uncommitted changes found, commit them before exiting(yes/no/cancel)?</td>
</tr>
<tr>
<td>Example: RP/O/RSP0/CPU0:router(config-mpls-te)# end</td>
<td>(cancel):</td>
</tr>
<tr>
<td></td>
<td>• Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</td>
</tr>
</tbody>
</table>
### Purpose

**Command or Action**

```command
or
RP/0/RSP0/CPU0:router(config-mpls-te)# commit
```

**Purpose**

- Entering `no` exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
- Entering `cancel` leaves the router in the current configuration session without exiting or committing the configuration changes.
- Use the `commit` command to save the configuration changes to the running configuration file and remain within the configuration session.

### Step 5

**Command or Action**

```command
show mpls traffic-eng tunnels [auto-bw]
```

**Example:**

```command
RP/0/RSP0/CPU0:router# show mpls traffic tunnels auto-bw
```

**Purpose**

Displays information about MPLS-TE tunnels for the automatic bandwidth. The globally configured collection frequency is displayed.

### Related Topics

- MPLS-TE Automatic Bandwidth Overview, page 20
- Configure Automatic Bandwidth: Example, page 121

### Forcing the Current Application Period to Expire Immediately

Perform this task to force the current application period to expire immediately on the specified tunnel. The highest bandwidth is applied on the tunnel before waiting for the application period to end on its own.

### SUMMARY STEPS

1. `mpls traffic-eng auto-bw apply {all | tunnel-te tunnel-number}`
2. `show mpls traffic-eng tunnels [auto-bw]`

### DETAILED STEPS

**Command or Action**

```command
mpls traffic-eng auto-bw apply {all | tunnel-te tunnel-number}
```

**Example:**

```command
RP/0/RSP0/CPU0:router# mpls traffic-eng auto-bw apply tunnel-te 1
```

**Purpose**

Configures the highest bandwidth available on a tunnel without waiting for the current application period to end.

- `all`: Configures the highest bandwidth available instantly on all the tunnels.
- `tunnel-te`: Configures the highest bandwidth instantly to the specified tunnel. Range is from 0 to 65535.
### Configuring the Automatic Bandwidth Functions

Perform this task to configure the following automatic bandwidth functions:

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Application frequency</strong></td>
<td>Configures the application frequency in which a tunnel bandwidth is updated by the automatic bandwidth.</td>
</tr>
<tr>
<td><strong>Bandwidth collection</strong></td>
<td>Configures only the bandwidth collection.</td>
</tr>
<tr>
<td><strong>Bandwidth parameters</strong></td>
<td>Configures the minimum and maximum automatic bandwidth to set on a tunnel.</td>
</tr>
<tr>
<td><strong>Adjustment threshold</strong></td>
<td>Configures the adjustment threshold for each tunnel.</td>
</tr>
<tr>
<td><strong>Overflow detection</strong></td>
<td>Configures the overflow detection for each tunnel.</td>
</tr>
</tbody>
</table>

### SUMMARY STEPS

1. `configure`
2. `interface tunnel-te tunnel-id`
3. `auto-bw`
4. `application minutes`
5. `bw-limit {min bandwidth} {max bandwidth}`
6. `adjustment-threshold percentage [min minimum-bandwidth]`
7. `overflow threshold percentage [min bandwidth] limit limit`
8. Use one of the following commands:
   - `end`
   - `commit`
9. `show mpls traffic-eng tunnels [auto-bw]`
## DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td><code>configure</code></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router# configure</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td><code>interface tunnel-te tunnel-id</code></td>
<td>Configures an MPLS-TE tunnel interface and enables traffic engineering on a particular interface on the originating node.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config)# interface tunnel-te 6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-if)#</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td><code>auto-bw</code></td>
<td>Configures automatic bandwidth on a tunnel interface and enters MPLS-TE automatic bandwidth interface configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-if)# auto-bw</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-if-tunte-autobw)#</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Step 4</td>
<td><code>application minutes</code></td>
<td>Configures the application frequency in minutes for the applicable tunnel.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-if-tunte-autobw)# application 1000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Step 5</td>
<td><code>bw-limit {min bandwidth} {max bandwidth}</code></td>
<td>Configures the minimum and maximum automatic bandwidth set on a tunnel.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-if-tunte-autobw)# bw-limit min 30 max 80</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Step 6</td>
<td><code>adjustment-threshold percentage [min minimum-bandwidth]</code></td>
<td>Configures the tunnel bandwidth change threshold to trigger an adjustment.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-if-tunte-autobw)# adjustment-threshold 50 min 800</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Purpose

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>the largest sample is higher or lower than the current tunnel bandwidth. Range is from 10 to 4294967295 kilobits per second (kbps). The default value is 10 kbps.</td>
</tr>
</tbody>
</table>

### Step 7

**overflow threshold** `percentage [min bandwidth] limit limit`

Configure the tunnel overflow detection.

- **percentage** Bandwidth change percent to trigger an overflow. Range is from 1 to 100 percent.
- **limit** Configures the number of consecutive collection intervals that exceeds the threshold. The bandwidth overflow triggers an early tunnel bandwidth update. Range is from 1 to 10 collection periods. The default value is none.
- **min** Configures the bandwidth change value in kbps to trigger an overflow. Range is from 10 to 4294967295. The default value is 10.

**Example:**

```
RP/0/RSP0/CPU0:router(config-if-tunte-autobw)#
overflow threshold 100 limit 1
```

### Step 8

Use one of the following commands:

- **end**
- **commit**

Saves configuration changes.

**Example:**

```
RP/0/RSP0/CPU0:router(config-if-tunte-autobw)#
end
```

- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
- Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
- Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.

- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

### Step 9

**show mpls traffic-eng tunnels [auto-bw]**

Displays the MPLS-TE tunnel information only for tunnels in which the automatic bandwidth is enabled.

**Example:**

```
RP/0/RSP0/CPU0:router# show mpls traffic-eng
```
Configuring the Shared Risk Link Groups

To activate the MPLS traffic engineering SRLG feature, you must configure the SRLG value of each link that has a shared risk with another link.

### Configuring the SRLG Values of Each Link that has a Shared Risk with Another Link

Perform this task to configure the SRLG value for each link that has a shared risk with another link.

#### Note
You can configure up to 30 SRLGs per interface.

#### SUMMARY STEPS

1. `configure`
2. `srlg`
3. `interface type interface-path-id`
4. `value value`
5. Use one of the following commands:
   - `end`
   - `commit`
6. `show srlg interface type interface-path-id`
7. `show srlg`

#### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td><code>configure</code></td>
</tr>
</tbody>
</table>

**Example:**

```
RP/0/RSP0/CPU0:router# configure
```
### Purpose

**Command or Action**  
**srlg**

**Example:**
RP/0/RSP0/CPU0:router(config)# srlg

**Purpose:** Configures SRLG configuration commands on a specific interface configuration mode and assigns this SRLG a value.

**Step 2**

### Step 3

**Command or Action**  
**interface** *type interface-path-id*

**Example:**
RP/0/RSP0/CPU0:router(config-srlg)# interface POS 0/6/0/0

**Purpose:** Configures an interface type and path ID to be associated with an SRLG and enters SRLG interface configuration mode.

**Step 3**

### Step 4

**Command or Action**  
**value** *value*

**Example:**
RP/0/RSP0/CPU0:router(config-srlg-if)# value 0  
RP/0/RSP0/CPU0:router(config-srlg-if)# value 100  
RP/0/RSP0/CPU0:router(config-srlg-if)# value 200  
RP/0/RSP0/CPU0:router(config-srlg-if)# value 300

**Purpose:** Configures SRLG network values for a specific interface. Range is 0 to 4294967295.

**Note:** You can also set SRLG values on multiple interfaces including bundle interface.

**Step 4**

### Step 5

**Command or Action**  
**Use one of the following commands:**
- **end**
- **commit**

**Example:**
RP/0/RSP0/CPU0:router(config)# end
or
RP/0/RSP0/CPU0:router(config)# commit

**Purpose:** Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

  Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:

  - Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
  
  - Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
  
  - Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.

  - Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

**Step 5**

### Step 6

**Command or Action**  
**show srlg interface** *type interface-path-id*

**Example:**
RP/0/RSP0/CPU0:router# show srlg interface POS 0/6/0/0

**Purpose:** (Optional) Displays the SRLG values configured for a specific interface.

**Step 6**

### Step 7

**Command or Action**  
**show srlg**

**Example:**
RP/0/RSP0/CPU0:router# show srlg

**Purpose:** (Optional) Displays the SRLG values for all the configured interfaces.
Creating an Explicit Path With Exclude SRLG

Perform this task to create an explicit path with the exclude SRLG option.

**SUMMARY STEPS**

1. configure
2. explicit-path {identifier number [disable | index]}{ name explicit-path-name}
3. index 1 exclude-address 192.168.92.1
4. index 2 exclude-srlg 192.168.92.2
5. Use one of the following commands:
   - end
   - commit

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>configure</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example: RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td>explicit-path {identifier number [disable</td>
<td>index]}{ name explicit-path-name}</td>
</tr>
<tr>
<td></td>
<td>Example: RP/0/RSP0/CPU0:router(config)# explicit-path name backup-srlg</td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td>index 1 exclude-address 192.168.92.1</td>
<td>Specifies the IP address to be excluded from the explicit path.</td>
</tr>
<tr>
<td></td>
<td>Example: RP/0/RSP0/CPU0:router router(config-expl-path)# index 1 exclude-address 192.168.92.1</td>
<td></td>
</tr>
</tbody>
</table>
### Command or Action

<table>
<thead>
<tr>
<th>Step 4</th>
<th>index 2 exclude-srlg 192.168.92.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router(config-expl-path)# index 2 exclude-srlg 192.168.92.2</td>
</tr>
</tbody>
</table>

**Purpose**

Specifies the IP address to extract SRLGs to be excluded from the explicit path.

### Step 5

Use one of the following commands:

- **end**
- **commit**

**Example:**

RP/0/RSP0/CPU0:router(config)# end

or

RP/0/RSP0/CPU0:router(config)# commit

**Purpose**

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

  Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:

  - Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
  - Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
  - Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.

- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

### Related Topics

- MPLS Traffic Engineering Shared Risk Link Groups, page 26
- Configure the MPLS-TE Shared Risk Link Groups: Example, page 121

### Using Explicit Path With Exclude SRLG

Perform this task to use an explicit path with the exclude SRLG option on the static backup tunnel.
### SUMMARY STEPS

1. configure
2. mpls traffic-eng
3. interface type interface-path-id
4. backup-path tunnel-te tunnel-number
5. exit
6. exit
7. interface tunnel-te tunnel-id
8. ipv4 unnumbered type interface-path-id
9. path-option preference-priority \{ dynamic | explicit \{identifier | name explicit-path-name\} \}
10. destination ip-address
11. exit
12. Use one of the following commands:
   - end
   - commit
13. show run explicit-path name name
14. show mpls traffic-eng topology path destination name explicit-path name

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example: RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> mpls traffic-eng</td>
<td>Enters MPLS-TE configuration mode.</td>
</tr>
<tr>
<td>Example: RP/0/RSP0/CPU0:router(config)# mpls traffic-eng</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> interface type interface-path-id</td>
<td>Enables traffic engineering on a specific interface on the originating node.</td>
</tr>
<tr>
<td>Example: RP/0/RSP0/CPU0:router(config-mpls-te)# interface POS 0/6/0/0</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> backup-path tunnel-te tunnel-number</td>
<td>Configures an MPLS TE backup path for a specific interface.</td>
</tr>
<tr>
<td>Example: RP/0/RSP0/CPU0:router(config-mpls-te)# backup-path tunnel-te 2</td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
</tr>
<tr>
<td>Step 5</td>
<td><code>exit</code>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;<code>RP/0/RSP0/CPU0:router(config-mpls-te-if)# exit</code>&lt;br&gt;Exits the current configuration mode.</td>
</tr>
<tr>
<td>Step 6</td>
<td><code>exit</code>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;<code>RP/0/RSP0/CPU0:router(config-mpls-te)# exit</code>&lt;br&gt;Exits the current configuration mode.</td>
</tr>
<tr>
<td>Step 7</td>
<td><code>interface tunnel-tunnel-id</code>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;<code>RP/0/RSP0/CPU0:router(config)# interface tunnel-te 2</code>&lt;br&gt;Configures an MPLS-TE tunnel interface.</td>
</tr>
<tr>
<td>Step 8</td>
<td><code>ipv4 unnumbered type interface-path-id</code>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;<code>RP/0/RSP0/CPU0:router(config-if)# ipv4 unnumbered Loopback0</code>&lt;br&gt;Assigns a source address to set up forwarding on the new tunnel.</td>
</tr>
<tr>
<td>Step 9</td>
<td>`path-option preference-priority{ dynamic</td>
</tr>
<tr>
<td>Step 10</td>
<td><code>destination ip-address</code>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;<code>RP/0/RSP0/CPU0:router(config-if)# destination 192.168.92.125</code>&lt;br&gt;Assigns a destination address on the new tunnel.&lt;br&gt;• Destination address is the remote node’s MPLS-TE router ID.&lt;br&gt;• Destination address is the merge point between backup and protected tunnels.&lt;br&gt;<strong>Note</strong>&lt;br&gt;When you configure TE tunnel with multiple protection on its path and merge point is the same node for more than one protection, you must configure record-route for that tunnel.</td>
</tr>
<tr>
<td>Step 11</td>
<td><code>exit</code>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;<code>RP/0/RSP0/CPU0:router(config-if)# exit</code>&lt;br&gt;Exits the current configuration mode.</td>
</tr>
<tr>
<td>Step 12</td>
<td>Use one of the following commands:&lt;br&gt;Saves configuration changes.</td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>----------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>• end</td>
<td>• When you issue the <strong>end</strong> command, the system prompts you to commit changes:</td>
</tr>
<tr>
<td>• commit</td>
<td>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</td>
</tr>
<tr>
<td>Example:</td>
<td>◦ Entering <strong>yes</strong> saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config)# end or RP/0/RSP0/CPU0:router(config)# commit</td>
<td>◦ Entering <strong>no</strong> exits the configuration session and returns the router to EXEC mode without committing the configuration changes.</td>
</tr>
<tr>
<td></td>
<td>◦ Entering <strong>cancel</strong> leaves the router in the current configuration session without exiting or committing the configuration changes.</td>
</tr>
<tr>
<td></td>
<td>• Use the <strong>commit</strong> command to save the configuration changes to the running configuration file and remain within the configuration session.</td>
</tr>
</tbody>
</table>

**Step 13**

<table>
<thead>
<tr>
<th>show run explicit-path name name</th>
<th>Displays the SRLG values that are configured for the link.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example:</td>
<td><strong>RP/0/RSP0/CPU0:router# show run explicit-path name backup-srlg</strong></td>
</tr>
</tbody>
</table>

**Step 14**

<table>
<thead>
<tr>
<th>show mpls traffic-eng topology path destination name explicit-path name</th>
<th>Displays the SRLG values that are configured for the link.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example:</td>
<td><strong>RP/0/RSP0/CPU0:router# show mpls traffic-eng topology path destination 192.168.92.125 explicit-path backup-srlg</strong></td>
</tr>
</tbody>
</table>

**Related Topics**

- [MPLS Traffic Engineering Shared Risk Link Groups](#), page 26
- [Configure the MPLS-TE Shared Risk Link Groups: Example](#), page 121

**Creating a Link Protection on Backup Tunnel with SRLG Constraint**

Perform this task to create an explicit path with the exclude SRLG option on the static backup tunnel.
SUMMARY STEPS

1. configure
2. mpls traffic-eng
3. interface type interface-path-id
4. backup-path tunnel-te tunnel-number
5. exit
6. exit
7. interface tunnel-tunnel-id
8. ipv4 unnumbered type interface-path-id
9. path-option preference-priority { dynamic | explicit {identifier | name explicit-path-name}}
10. destination ip-address
11. exit
12. explicit-path {identifier number [disable | index]} { name explicit-path-name}
13. index 1 exclude-srlg 192.168.92.2
14. Use one of the following commands:
   • end
   • commit
15. show mpls traffic-eng tunnelstunnel-number detail

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> mpls traffic-eng</td>
<td>Enters MPLS-TE configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> RP/0/RSP0/CPU0:router(config)# mpls traffic-eng</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> interface type interface-path-id</td>
<td>Enables traffic engineering on a particular interface on the originating node.</td>
</tr>
<tr>
<td><strong>Example:</strong> RP/0/RSP0/CPU0:router(config-mpls-te)# interface POS 0/6/0/0</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> backup-path tunnel-te tunnel-number</td>
<td>Sets the backup path to the primary tunnel outgoing interface.</td>
</tr>
<tr>
<td><strong>Example:</strong> RP/0/RSP0/CPU0:router(config-mpls-te)# backup-path tunnel-te 2</td>
<td></td>
</tr>
<tr>
<td>Step</td>
<td>Command or Action</td>
</tr>
<tr>
<td>------</td>
<td>------------------</td>
</tr>
<tr>
<td>5</td>
<td>exit</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> RP/0/RSP0/CPU0:router(config-mpls-te-if)# exit</td>
</tr>
<tr>
<td>6</td>
<td>exit</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> RP/0/RSP0/CPU0:router(config-mpls-te)# exit</td>
</tr>
<tr>
<td>7</td>
<td>interface tunnel-te <code>tunnel-id</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> RP/0/RSP0/CPU0:router(config)# interface tunnel-te 2</td>
</tr>
<tr>
<td>8</td>
<td>ipv4 unnumbered <code>type interface-path-id</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> RP/0/RSP0/CPU0:router(config-if)# ipv4 unnumbered Loopback0</td>
</tr>
</tbody>
</table>
| 9    | path-option `preference-priority` `dynamic | explicit` {identifier | name `explicit-path-name`} | Sets the path option to explicit with a given name (previously configured) and assigns the path ID. Identifier range is from 1 to 4294967295.  
|      | **Example:** RP/0/RSP0/CPU0:router(config-if)# path-option 1 explicit name backup-srlg |         |
|      | **Note** You can use the dynamic option to dynamically assign a path. |         |
| 10   | destination `ip-address` | Assigns a destination address on the new tunnel.  
|      | • Destination address is the remote node’s MPLS-TE router ID.  
|      | • Destination address is the merge point between backup and protected tunnels.  
|      | **Note** When you configure TE tunnel with multiple protection on its path and merge point is the same node for more than one protection, you must configure record-route for that tunnel. |         |
| 11   | exit             | Exits the current configuration mode. |
|      | **Example:** RP/0/RSP0/CPU0:router(config-if)# exit |         |
| 12   | explicit-path `identifier number` `disable | index` `name` `explicit-path-name` | Enters the explicit path configuration mode. Identifier range is 1 to 65535. |
|      | **Example:** RP/0/RSP0/CPU0:router(config)# explicit-path name backup-srlg-nodep |         |
### Purpose

**Command or Action**

| Step 13 | index 1 exclude-srlg 192.168.92.2 |

**Example:**

RP/O/RSP0/CPU0:router# router(config-if)# index 1 exclude-srlg 192.168.92.2

**Purpose:** Specifies the protected link IP address to get SRLGs to be excluded from the explicit path.

| Step 14 | Use one of the following commands: |

- end
- commit

**Example:**

RP/O/RSP0/CPU0:router(config)# end
RP/O/RSP0/CPU0:router(config)# commit

**Purpose:** Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

  Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:

  - Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
  - Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
  - Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.

  - Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

| Step 15 | show mpls traffic-eng tunnels tunnel-number detail |

**Example:**

RP/O/RSP0/CPU0:router# show mpls traffic-eng tunnels 2 detail

**Purpose:** Display the tunnel details with SRLG values that are configured for the link.

### Related Topics

- MPLS Traffic Engineering Shared Risk Link Groups, page 26
- Configure the MPLS-TE Shared Risk Link Groups: Example, page 121

### Creating a Node Protection on Backup Tunnel with SRLG Constraint

Perform this task to configure node protection on backup tunnel with SRLG constraint.
SUMMARY STEPS

1. configure
2. mpls traffic-eng
3. interface type interface-path-id
4. backup-path tunnel-te tunnel-number
5. exit
6. exit
7. interface tunnel-te tunnel-id
8. ipv4 unnumbered type interface-path-id
9. path-option preference-priority{ dynamic | explicit {identifier | name explicit-path-name}}
10. destination ip-address
11. exit
12. explicit-path {identifier number [disable | index]}{ name explicit-path-name}
13. index 1 exclude-address 192.168.92.1
14. index 2 exclude-srlg 192.168.92.2
15. Use one of the following commands:
   * end
   * commit
16. show mpls traffic-eng tunnels topology path destination ip-address explicit-path-name name

DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>configure</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td>mpls traffic-eng</td>
<td>Enters MPLS-TE configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router(config)# mpls traffic-eng</td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td>interface type interface-path-id</td>
<td>Enables traffic engineering on a particular interface on the originating node.</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)# interface POS 0/6/0/0</td>
<td></td>
</tr>
<tr>
<td>Step 4</td>
<td>backup-path tunnel-te tunnel-number</td>
<td>Sets the backup path for the primary tunnel outgoing interface.</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RSP0/CPU0:router(config-mpls-te)# backup-path tunnel-te 2</td>
<td></td>
</tr>
<tr>
<td>Step</td>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------</td>
<td>-------------------</td>
<td>---------</td>
</tr>
<tr>
<td>5</td>
<td>exit</td>
<td>Exits the current configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example: RP/0/RSP0/CPU0:router(config-mpls-te-if)# exit</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>exit</td>
<td>Exits the current configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example: RP/0/RSP0/CPU0:router(config-mpls-te)# exit</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>interface tunnel-te tunnel-id</td>
<td>Configures an MPLS-TE tunnel interface.</td>
</tr>
<tr>
<td></td>
<td>Example: RP/0/RSP0/CPU0:router(config)# interface tunnel-te 2</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>ipv4 unnumbered type interface-path-id</td>
<td>Assigns a source address to set up forwarding on the new tunnel.</td>
</tr>
<tr>
<td></td>
<td>Example: RP/0/RSP0/CPU0:router(config-if)# ipv4 unnumbered Loopback0</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>path-option preference-priority { dynamic</td>
<td>explicit</td>
</tr>
<tr>
<td></td>
<td>Example: RP/0/RSP0/CPU0:router(config-if)# path-option 1 explicit name backup-srlg</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Note You can use the dynamic option to dynamically assign path.</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>destination ip-address</td>
<td>Assigns a destination address on the new tunnel.</td>
</tr>
<tr>
<td></td>
<td>Example: RP/0/RSP0/CPU0:router(config-if)# destination 192.168.92.125</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Destination address is the remote node’s MPLS-TE router ID.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Destination address is the merge point between backup and protected tunnels.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Note When you configure TE tunnel with multiple protection on its path and merge point is the same node for more than one protection, you must configure record-route for that tunnel.</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>exit</td>
<td>Exits the current configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example: RP/0/RSP0/CPU0:router(config-if)# exit</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>explicit-path {identifier number {disable</td>
<td>index}</td>
</tr>
<tr>
<td></td>
<td>Example: RP/0/RSP0/CPU0:router(config)# explicit-path name backup-srlg-nodep</td>
<td></td>
</tr>
</tbody>
</table>
### Command or Action

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 13</strong></td>
<td>index 1 exclude-address 192.168.92.1</td>
<td>Specifies the protected node IP address to be excluded from the explicit path.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RSP0/CPU0:router# router(config-if)# index 1 exclude-address 192.168.92.1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 14</strong></td>
<td>index 2 exclude-srlg 192.168.92.2</td>
<td>Specifies the protected link IP address to get SRLGs to be excluded from the explicit path.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/RSP0/CPU0:router(config-if)# index 2 exclude-srlg 192.168.192.2</td>
<td></td>
</tr>
</tbody>
</table>
| **Step 15** | Use one of the following commands:  
- end  
- commit | Saves configuration changes.  
- When you issue the **end** command, the system prompts you to commit changes:  
  Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:  
  ◦ Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.  
  ◦ Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.  
  ◦ Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.  
- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session. |
| **Example:** | RP/0/RSP0/CPU0:router# end  
or  
RP/0/RSP0/CPU0:router# commit | |
| **Step 16** | show mpls traffic-eng tunnels topology path destination ip-address explicit-path-name name | Displays the path to the destination with the constraint specified in the explicit path. |
| **Example:** | RP/0/RSP0/CPU0:router# show mpls traffic-eng tunnels topology path destination 192.168.92.125 explicit-path-name backup-srlg-nodep | |

### Related Topics
- [MPLS Traffic Engineering Shared Risk Link Groups](#), page 26
- [Configure the MPLS-TE Shared Risk Link Groups: Example](#), page 121
Configuring Point-to-Multipoint TE

You must enable multicast routing on the edge router before performing Point-to-Multipoint (P2MP) TE configurations. To configure Point-to-Multipoint TE, perform these procedures:

Enabling Multicast Routing on the Router

Perform this task to enable multicast routing on the router to configure P2MP tunnels.

Before You Begin

• To configure Point-to-Multipoint (P2MP) tunnels, you must enable multicast routing on the router.
• The customer-facing interface must enable multicast.

SUMMARY STEPS

1. configure
2. multicast-routing
3. address-family {ipv4 | ipv6 }
4. interface tunnel-mte tunnel-id
5. enable
6. exit
7. interface type interface-path-id
8. enable
9. Use one of these commands:
   • end
   • commit
10. show pim ipv6 interface type interface-path-id

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 configure</td>
<td>Enters global configuration mode.</td>
</tr>
</tbody>
</table>
| Example:  
RP/0/RSP0/CPU0:router# configure |                                               |
| Step 2 multicast-routing   | Enters multicast routing configuration mode. |
| Example:  
RP/0/RSP0/CPU0:router(config)# multicast-routing |                                               |
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 3</strong> address-family {ipv4</td>
<td>ipv6 }</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>RP/0/RSP0/CPU0:router(config-mcast)# address-family ipv6</code></td>
<td></td>
</tr>
<tr>
<td><code>RP/0/RSP0/CPU0:router(config-mcast-default-ipv6)#</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> interface tunnel-mte <code>tunnel-id</code></td>
<td>Configures an MPLS-TE P2MP tunnel interface.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>RP/0/RSP0/CPU0:router(config-mcast-default-ipv6)# interface tunnel-mte 1</code></td>
<td></td>
</tr>
<tr>
<td><code>RP/0/RSP0/CPU0:router(config-mcast-default-ipv6-if)#</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> enable</td>
<td>Enables multicast routing on the tunnel-mte interface.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>RP/0/RSP0/CPU0:router(config-mcast-default-ipv6-if)# enable</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong> exit</td>
<td>Exits the current configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>RP/0/RSP0/CPU0:router(config-mcast-default-ipv6-if)# exit</code></td>
<td></td>
</tr>
<tr>
<td><code>RP/0/RSP0/CPU0:router(config-mcast-default-ipv6)#</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 7</strong> interface type <code>interface-path-id</code></td>
<td>Configures multicast routing on the GigabitEthernet interface.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>RP/0/RSP0/CPU0:router(config-mcast-default-ipv6)# interface GigabitEthernet0/2/0/3</code></td>
<td></td>
</tr>
<tr>
<td><code>RP/0/RSP0/CPU0:router(config-mcast-default-ipv6-if)#</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 8</strong> enable</td>
<td>Enables multicast routing on the GigabitEthernet interface.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>RP/0/RSP0/CPU0:router(config-mcast-default-ipv6-if)# enable</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 9</strong></td>
<td>Saves configuration changes.</td>
</tr>
<tr>
<td><strong>Use one of these commands:</strong></td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>------------------</td>
<td>---------</td>
</tr>
</tbody>
</table>
| • end            | • When you issue the **end** command, the system prompts you to commit changes: Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:  
  ◦ Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.  
  ◦ Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.  
  ◦ Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.  
| • commit         | • Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session. |

**Example:**

RP/0/RSP0/CPU0:router(config-mcast-default-ipv6-if)#
  
  or
  
  RP/0/RSP0/CPU0:router(config-mcast-default-ipv6-if)#

  **Step 10**

  **show pim ipv6 interface** *type interface-path-id*

  **Example:**

  RP/0/RSP0/CPU0:router# **show pim ipv6 interface tunnel-mte 1**

  Displays the output for the P2MP-TE tunnel interface that has IPv6 multicast enabled.

**Related Topics**

• Configuring the Static Group for the Point-to-Multipoint Interface, page 104

**Configuring the Static Group for the Point-to-Multipoint Interface**

Perform this task to configure the static group on the Point-to-Multipoint (P2MP) interface to forward specified multicast traffic over P2MP LSP.
SUMMARY STEPS

1. configure
2. router mld
3. vrf vrf-name
4. interface tunnel-mte tunnel-id
5. static-group group-address
6. Use one of these commands:
   • end
   • commit

7. show mrrib ipv6 route source-address

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> router mld</td>
<td>Enters router MLD configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config)# router mld</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mld)#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> vrf vrf-name</td>
<td>Configures a virtual private network (VRF) instance.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mld)# vrf default</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mld-default)#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> interface tunnel-mte tunnel-id</td>
<td>Configures an MPLS-TE P2MP tunnel interface.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mld-default)# interface tunnel-mte 1</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mld-default-if)#</td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Step 5</strong> static-group group-address</td>
<td>Configures the multicast group address in the Source-Specific Multicast (SSM) address range (ff35::/16) for the IPv6 address prefix.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mld-default-if)# static-group ff35::1 2000::1</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 6</th>
<th>Use one of these commands:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• end</td>
<td>Saves configuration changes.</td>
</tr>
<tr>
<td>• commit</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mld-default-if)# end</td>
<td></td>
</tr>
<tr>
<td>or</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-mld-default-if)# commit</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 7</th>
<th>show mrib ipv6 route source-address</th>
<th>Verifies the multicast static mapping.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router# show mrib ipv6 route ff35::1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Related Topics**
- Enabling Multicast Routing on the Router, page 102

**Configuring Destinations for the Tunnel Interface**

These variations are listed to ensure that the destination and path option configurations are separate from the tunnel interface.

- Different path option is used for different destinations. This task shows three destinations.
- Explicit path option is based on an ID or a name.
• Default path option is similar to the Point-to-Point (P2P) LSP.

**Before You Begin**

These prerequisites are required to configure destinations for the tunnel interface.

• Multicast routing must be enabled on both the tunnel-mte interface and customer-facing interface from the source.

• Static-group must be configured on the tunnel-mte interface to forward specified multicast traffic over P2MP LSP.

**SUMMARY STEPS**

1. configure
2. interface tunnel-mte tunnel-id
3. destination ip-address
4. path-option preference-priority explicit identifier path-number
5. path-option preference-priority dynamic
6. exit
7. destination ip-address
8. path-option preference-priority explicit name pathname
9. path-option preference-priority dynamic
10. exit
11. destination ip-address
12. path-option preference-priority explicit name pathname [verbatim]
13. Use one of these commands:
   • end
   • commit
14. show mpls traffic-eng tunnels [brief] [p2mp tunnel-number]

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**  
  configure | Enters global configuration mode. |
| **Example:**  
  RP/0/RSP0/CPU0:router# configure | |
| **Step 2**  
  interface tunnel-mte tunnel-id | Configures an MPLS-TE P2MP tunnel interface. |
| **Example:**  
  RP/0/RSP0/CPU0:router(config)# interface | |
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>tunnel-mte 10</strong>&lt;br&gt; RP/0/RSP0/CPU0:router(config-if)#</td>
<td>Sets the destination address for tunnel-mte 10 to 172.16.255.1. This destination uses the explicit path identified by explicit path ID 10. If destination 172.16.255.1 cannot come with explicit path ID 10, the fall back path option is dynamic.</td>
</tr>
<tr>
<td><strong>Step 3</strong>&lt;br&gt;destination <em>ip-address</em>&lt;br&gt;Example:&lt;br&gt;RP/0/RSP0/CPU0:router(config-if)# destination 172.16.255.1&lt;br&gt;RP/0/RSP0/CPU0:router(config-if-p2mp-dest)#</td>
<td>Configures the path number of the IP explicit path.</td>
</tr>
<tr>
<td><strong>Step 4</strong>&lt;br&gt;path-option <em>preference-priority</em> explicit identifier <em>path-number</em>&lt;br&gt;Example:&lt;br&gt;RP/0/RSP0/CPU0:router(config-if-p2mp-dest)# path-option 1 explicit identifier 10</td>
<td>Specifies that label switched paths (LSP) are dynamically calculated.</td>
</tr>
<tr>
<td><strong>Step 5</strong>&lt;br&gt;path-option <em>preference-priority</em> dynamic&lt;br&gt;Example:&lt;br&gt;RP/0/RSP0/CPU0:router(config-if-p2mp-dest)# path-option 2 dynamic</td>
<td>Exits the current configuration mode.</td>
</tr>
<tr>
<td><strong>Step 6</strong>&lt;br&gt;exit&lt;br&gt;Example:&lt;br&gt;RP/0/RSP0/CPU0:router(config-if-p2mp-dest)# exit&lt;br&gt;RP/0/RSP0/CPU0:router(config-if)#</td>
<td>Sets the destination address for tunnel-mte 10 to 172.16.255.2.</td>
</tr>
<tr>
<td><strong>Step 7</strong>&lt;br&gt;destination <em>ip-address</em>&lt;br&gt;Example:&lt;br&gt;RP/0/RSP0/CPU0:router(config-if)# destination 172.16.255.2&lt;br&gt;RP/0/RSP0/CPU0:router(config-if-p2mp-dest)#</td>
<td>Specifies the path name of the IP explicit path. Destination 172.16.255.2 uses the explicit path that is identified by the explicit path name &quot;how-to-get-to-172.16.255.2.&quot;</td>
</tr>
<tr>
<td><strong>Step 8</strong>&lt;br&gt;path-option <em>preference-priority</em> explicit name <em>pathname</em>&lt;br&gt;Example:&lt;br&gt;RP/0/RSP0/CPU0:router(config-if-p2mp-dest)# path-option 1 explicit name how-to-get-to-172.16.255.2</td>
<td></td>
</tr>
<tr>
<td>Step</td>
<td>Command or Action</td>
</tr>
<tr>
<td>-------</td>
<td>--------------------------------------------------------</td>
</tr>
<tr>
<td>Step 9</td>
<td>path-option <em>preference-priority</em> dynamic</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-if-p2mp-dest)# path-option 2 dynamic</td>
</tr>
<tr>
<td>Step 10</td>
<td>exit</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-if-p2mp-dest)# exit</td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-if)#</td>
</tr>
<tr>
<td>Step 11</td>
<td>destination <em>ip-address</em></td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-if)# destination 172.16.255.3 #</td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-if-p2mp-dest)#</td>
</tr>
<tr>
<td>Step 12</td>
<td>path-option <em>preference-priority</em> explicit name <em>pathname</em> [verbatim]</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-if-p2mp-dest)# path-option 1 explicit name how-to-get-to-172.16.255.3 verbatim</td>
</tr>
<tr>
<td>Step 13</td>
<td>Use one of these commands:</td>
</tr>
<tr>
<td></td>
<td>• end</td>
</tr>
<tr>
<td></td>
<td>• commit</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-if-p2mp-dest)# end</td>
</tr>
<tr>
<td></td>
<td>or</td>
</tr>
<tr>
<td></td>
<td>RP/0/RSP0/CPU0:router(config-if-p2mp-dest)# commit</td>
</tr>
</tbody>
</table>

- When you issue the `end` command, the system prompts you to commit changes:

  Uncommitted changes found, commit them before exiting(yes/no/cancel)?
  [cancel]:

  - Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
  - Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
  - Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
### Configuring Point-to-Multipoint TE

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show mpls traffic-eng tunnels [brief] [p2mp tunnel-number]</td>
<td>Displays the brief summary of the P2MP tunnel status and configuration.</td>
</tr>
</tbody>
</table>

**Example:**

```
RP/0/RSP0/CPU0:router# show mpls traffic-eng tunnels brief p2mp 10
```

#### Related Topics
- Enabling Multicast Routing on the Router, page 102
- Configuring the Static Group for the Point-to-Multipoint Interface, page 104

### Disabling Destinations

Perform this task to disable the given destination for the Point-to-Multipoint (P2MP) tunnel interface.

#### SUMMARY STEPS

1. `configure`
2. `interface tunnel-mte tunnel-id`
3. `ipv4 unnumbered type interface-path-id`
4. `destination ip-address`
5. `disable`
6. `path-option preference-priority dynamic`
7. `path-option preference-priority explicit name pathname`
8. Use one of these commands:
   - `end`
   - `commit`

#### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure</td>
<td>Enters global configuration mode.</td>
</tr>
</tbody>
</table>

**Example:**

```
RP/0/RSP0/CPU0:router# configure
```
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>interface tunnel-mte</strong> <em>tunnel-id</em></td>
</tr>
<tr>
<td><em>Example:</em></td>
<td>RP/0/RSP0/CPU0:router(config)# <code>interface tunnel-mte 101</code> RP/0/RSP0/CPU0:router(config-if)#</td>
</tr>
<tr>
<td><strong>Purpose:</strong></td>
<td>Configures an MPLS-TE P2MP tunnel interface.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td><strong>ipv4 unnumbered</strong> <em>type interface-path-id</em></td>
</tr>
<tr>
<td><em>Example:</em></td>
<td>RP/0/RSP0/CPU0:router(config-if)# <code>ipv4 unnumbered Loopback0</code></td>
</tr>
<tr>
<td><strong>Purpose:</strong></td>
<td>Assigns a source address so that forwarding can be performed on the new tunnel. Loopback is commonly used as the interface type.</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td><strong>destination</strong> <em>ip-address</em></td>
</tr>
<tr>
<td><em>Example:</em></td>
<td>RP/0/RSP0/CPU0:router(config-if)# <code>destination 140.140.140.140</code> RP/0/RSP0/CPU0:router(config-if-p2mp-dest)#</td>
</tr>
<tr>
<td><strong>Purpose:</strong></td>
<td>Sets the destination address for tunnel-mte 10 to 140.140.140.140.</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td><strong>disable</strong></td>
</tr>
<tr>
<td><em>Example:</em></td>
<td>RP/0/RSP0/CPU0:router(config-if-p2mp-dest)#<code>disable</code></td>
</tr>
<tr>
<td><strong>Purpose:</strong></td>
<td>Disables destination 140.140.140.140 for tunnel-mte 10.</td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td><strong>path-option</strong> <em>preference-priority</em> <em>dynamic</em></td>
</tr>
<tr>
<td><em>Example:</em></td>
<td>RP/0/RSP0/CPU0:router(config-if-p2mp-dest)#<code>path-option 1 dynamic</code></td>
</tr>
<tr>
<td><strong>Purpose:</strong></td>
<td>Specifies that label switched paths (LSP) are dynamically calculated.</td>
</tr>
<tr>
<td><strong>Step 7</strong></td>
<td><strong>path-option</strong> <em>preference-priority</em> <em>explicit name</em> <em>pathname</em></td>
</tr>
<tr>
<td><em>Example:</em></td>
<td>RP/0/RSP0/CPU0:router(config-if-p2mp-dest)#<code>path-option 2 explicit name to4</code></td>
</tr>
<tr>
<td><strong>Purpose:</strong></td>
<td>Specifies that destination 140.140.140.140 uses the explicit path identified by the explicit path name “to4.”</td>
</tr>
<tr>
<td><strong>Step 8</strong></td>
<td>Use one of these commands:</td>
</tr>
</tbody>
</table>
| *end* | Saves configuration changes.
| *commit* | *When you issue the** *end* **command, the system prompts you to commit changes:*
| | Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: |
Purpose

Command or Action | Purpose
--- | ---
Example: | • Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
RP/0/RSP0/CPU0:router(config-if-p2mp-dest)# end or RP/0/RSP0/CPU0:router(config-if-p2mp-dest)# commit | • Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
• Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Logging Per Destinations for Point-to-Multipoint

Perform this task to log destinations for Point-to-Multipoint (P2MP).

**SUMMARY STEPS**

1. **configure**
2. **interface tunnel-mte** *tunnel-id*  
3. **ipv4 unnumbered** *type interface-path-id*  
4. **destination** *ip-address*  
5. **logging events lsp-status state**  
6. **logging events lsp-status reroute**  
7. **path-option** *preference-priority explicit name* *pathname*  
8. **exit**  
9. **fast-reroute**  
10. Use one of these commands:  
    - **end**  
    - **commit**  
11. **show mpls traffic-eng tunnels [p2mp]**
## DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>configure</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong>&lt;br&gt;RP/0/RSP0/CPU0:router# configure</td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td>interface tunnel-mte tunnel-id</td>
<td>Configures an MPLS-TE P2MP tunnel interface.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong>&lt;br&gt;RP/0/RSP0/CPU0:router(config)# interface tunnel-mte 1000&lt;br&gt;RP/0/RSP0/CPU0:router(config-if)#</td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td>ipv4 unnumbered type interface-path-id</td>
<td>Configures the MPLS-TE tunnel to use the IPv4 address on loopback interface 0.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong>&lt;br&gt;RP/0/RSP0/CPU0:router(config-if)# ipv4 unnumbered loopback0</td>
<td></td>
</tr>
<tr>
<td>Step 4</td>
<td>destination ip-address</td>
<td>Sets the destination address for tunnel-mte from 1000 to 100.0.0.3.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong>&lt;br&gt;RP/0/RSP0/CPU0:router(config-if)# destination 100.0.0.3&lt;br&gt;RP/0/RSP0/CPU0:router(config-if-p2mp-dest)#</td>
<td></td>
</tr>
<tr>
<td>Step 5</td>
<td>logging events lsp-status state</td>
<td>Sends out the log message when the tunnel LSP goes up or down when the software is enabled.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong>&lt;br&gt;RP/0/RSP0/CPU0:router(config-if-p2mp-dest)# logging events lsp-status state</td>
<td></td>
</tr>
<tr>
<td>Step 6</td>
<td>logging events lsp-status reroute</td>
<td>Sends out the log message when the tunnel LSP is rerouted due to an FRR event when the software is enabled.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong>&lt;br&gt;RP/0/RSP0/CPU0:router(config-if-p2mp-dest)# logging events lsp-status reroute</td>
<td></td>
</tr>
<tr>
<td>Step 7</td>
<td>path-option preference-priority explicit name pathname</td>
<td>Specifies the path name of the IP explicit path. Destination 100.0.0.3 uses the explicit path that is identified by the explicit path name &quot;path123.&quot;</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong>&lt;br&gt;RP/0/RSP0/CPU0:router(config-if-p2mp-dest)#</td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
<td></td>
</tr>
<tr>
<td>-------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>path-option 1 explicit name path123</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Step 8</strong> exit</td>
<td>Exits the current configuration mode.</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-if-p2mp-dest)# exit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-if)#</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Step 9</strong> fast-reroute</td>
<td>Enables fast-reroute (FRR) protection for a P2MP TE tunnel.</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-if)# fast-reroute</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Step 10</strong> Use one of these commands:</td>
<td>Saves configuration changes.</td>
<td></td>
</tr>
<tr>
<td>• end</td>
<td>• When you issue the <strong>end</strong> command, the system prompts you to commit changes:</td>
<td></td>
</tr>
<tr>
<td>• commit</td>
<td>Uncommitted changes found, commit them before exiting(yes/no/cancel)?</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>[cancel]:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-if)# end</td>
<td>◦ Entering <strong>yes</strong> saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</td>
<td></td>
</tr>
<tr>
<td>or</td>
<td>◦ Entering <strong>no</strong> exits the configuration session and returns the router to EXEC mode without committing the configuration changes.</td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router(config-if)# commit</td>
<td>◦ Entering <strong>cancel</strong> leaves the router in the current configuration session without exiting or committing the configuration changes.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 11</strong> show mpls traffic-eng tunnels [p2mp]</td>
<td>Displays the information for all P2MP tunnels.</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router# show mpls traffic-eng</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tunnels p2mp</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Configuration Examples for Cisco MPLS-TE

These configuration examples are used for MPLS-TE:

Build MPLS-TE Topology and Tunnels: Example

The following examples show how to build an OSPF and IS-IS topology:

(OSPF)
...
configure
  mpls traffic-eng
  interface pos 0/6/0/0
  router id loopback 0
  router ospf 1
  router-id 192.168.25.66
  area 0
  interface pos 0/6/0/0
  interface loopback 0
  mpls traffic-eng router-id loopback 0
  mpls traffic-eng area 0
  rsvp
  interface pos 0/6/0/0
  bandwidth 100
  commit
  show mpls traffic-eng topology
  show mpls traffic-eng link-management advertisement
  !
(IS-IS)
...
configure
  mpls traffic-eng
  interface pos 0/6/0/0
  router id loopback 0
  router isis lab
  address-family ipv4 unicast
  mpls traffic-eng level 2
  mpls traffic-eng router-id Loopback 0
  !
  interface POS0/0/0/0
  path-option l dynamic
  address-family ipv4 unicast
  !

The following example shows how to configure tunnel interfaces:

interface tunnel-te1
  destination 192.168.92.125
  ipv4 unnumbered loopback 0
  path-option l dynamic
  bandwidth 100
  commit
  show mpls traffic-eng tunnels
  show ipv4 interface brief
  show mpls traffic-eng link-management admission-control
  !
  interface tunnel-te1
  autoroute announce
  route ipv4 192.168.12.52/32 tunnel-te1
  commit
  ping 192.168.12.52
  show mpls traffic autoroute
  !
  interface tunnel-te1
  fast-reroute
mpls traffic-eng interface pos 0/6/0/0
backup-path tunnel-te 2
interface tunnel-te2
backup-bw global-pool 5000
ipv4 unnumbered loopback 0
path-option 1 explicit name backup-path
destination 192.168.92.125
commit
show mpls traffic-eng tunnels backup
show mpls traffic-eng fast-reroute database
!
rserv
interface pos 0/6/0/0
bandwidth 100 150 sub-pool 50
interface tunnel-te1
bandwidth sub-pool 10
commit

Related Topics
• Building MPLS-TE Topology, page 30
• Creating an MPLS-TE Tunnel, page 33
• How MPLS-TE Works, page 3

Configure IETF DS-TE Tunnels: Example

The following example shows how to configure DS-TE:

rserv
interface pos 0/6/0/0
bandwidth rdm 100 150 bc1 50
mpls traffic-eng
ds-te mode ietf
interface tunnel-te 1
bandwidth 10 class-type 1
commit
configure
rserv interface 0/6/0/0
bandwidth mam max-reservable-bw 400 bc0 300 bc1 200
mpls traffic-eng
ds-te mode ietf
ds-te model mam
interface tunnel-te 1 bandwidth 10 class-type 1
commit

Related Topics
• Configuring a Prestandard DS-TE Tunnel, page 49
• Prestandard DS-TE Mode, page 8

Configure MPLS-TE and Fast-Reroute on OSPF: Example

CSPF areas are configured on a per-path-option basis. The following example shows how to use the traffic-engineering tunnels (tunnel-te) interface and the active path for the MPLS-TE tunnel:

configure
interface tunnel-te 0
path-option 1 explicit id 6 ospf 126 area 0
path-option 2 explicit name 234 ospf 3 area 7 verbatim
path-option 3 dynamic isis mtbf level 1 lockdown
Configure the Ignore IS-IS Overload Bit Setting in MPLS-TE: Example

This example shows how to configure the IS-IS overload bit setting in MPLS-TE:

This figure illustrates the IS-IS overload bit scenario:

Figure 9: IS-IS overload bit

Consider a MPLS TE topology in which usage of nodes that indicated an overload situation was restricted. In this topology, the router R7 exhibits overload situation and hence this node cannot be used during TE CSPF. To overcome this limitation, the IS-IS overload bit avoidance (OLA) feature was introduced. This feature allows network administrators to prevent RSVP-TE label switched paths (LSPs) from being disabled when a router in that path has its Intermediate System-to-Intermediate System (IS-IS) overload bit set.

The IS-IS overload bit avoidance feature is activated at router R1 using this command:

```
mpls traffic-eng path-selection ignore overload
```

Related Topics
- Configuring the Ignore Integrated IS-IS Overload Bit Setting in MPLS-TE, page 59
- Ignore Intermediate System-to-Intermediate System Overload Bit Setting in MPLS-TE, page 12

Configure Flexible Name-based Tunnel Constraints: Example

The following configuration shows the three-step process used to configure flexible name-based tunnel constraints.

```
R2
line console
  exec-timeout 0 0
  width 250
```
logging console debugging
explicit-path name mypath
index 1 next-address loose ipv4 unicast 3.3.3.3
explicit-path name ex_path1
index 10 next-address loose ipv4 unicast 2.2.2.2
index 20 next-address loose ipv4 unicast 3.3.3.3
interface Loopback0
ipv4 address 22.22.22.22 255.255.255.255
interface tunnel-te1
ipv4 unnumbered Loopback0
signalled-bandwidth 1000000
destination 3.3.3.3
affinity include green
affinity include yellow
affinity exclude white
affinity exclude orange
path-option 1 dynamic
router isis 1
is-type level-1
net 47.0001.0000.0000.0001.00
nsf cisco
address-family ipv4 unicast
metric-style wide
mpls traffic-eng level-1
mpls traffic-eng router-id Loopback0
interface Loopback0
passive
address-family ipv4 unicast
interface GigabitEthernet0/1/0/0
address-family ipv4 unicast
interface GigabitEthernet0/1/0/1
address-family ipv4 unicast
interface GigabitEthernet0/1/0/2
address-family ipv4 unicast
interface GigabitEthernet0/1/0/3
address-family ipv4 unicast
rsvp
interface GigabitEthernet0/1/0/0
bandwidth 1000000 1000000
interface GigabitEthernet0/1/0/1
bandwidth 1000000 1000000
interface GigabitEthernet0/1/0/2
bandwidth 1000000 1000000
interface GigabitEthernet0/1/0/3
bandwidth 1000000 1000000
mpls traffic-eng
interface GigabitEthernet0/1/0/0
attribute-names red purple
interface GigabitEthernet0/1/0/1
attribute-names red orange
interface GigabitEthernet0/1/0/2
attribute-names green purple
Configure an Interarea Tunnel: Example

The following configuration example shows how to configure a traffic engineering interarea tunnel.

```plaintext
configure
interface Tunnel-te1
ipv4 unnumbered Loopback0
destination 192.168.20.20
signalled-bandwidth 300
path-option 1 explicit name path-tunnel1
extent-path name path-tunnel1
next-address loose 192.168.40.40
next-address loose 192.168.60.60
next-address loose 192.168.20.20
```

Note

Specifying the tunnel tailend in the loosely routed path is optional.

Note

Generally for an interarea tunnel you should configure multiple loosely routed path options that specify different combinations of ABRs (for OSPF) or level-1-2 boundary routers (for IS-IS) to increase the likelihood that the tunnel is successfully signaled. In this simple topology there are no other loosely routed paths.

Configure Forwarding Adjacency: Example

The following configuration example shows how to configure an MPLS-TE forwarding adjacency on tunnel-te 68 with a holdtime value of 60:

```plaintext
configure
interface tunnel-te 68
forwarding-adjacency holdtime 60
commit
```
Configure PCE: Example

The following configuration example illustrates a PCE configuration:

```
configure
mpls traffic-eng
  interface pos 0/6/0/0
  pce address ipv4 192.168.25.66
  router id loopback 0
  router ospf 1
  router-id 192.168.25.66
  area 0
  interface pos 0/6/0/0
  interface loopback 0
  mpls traffic-eng router-id loopback 0
  mpls traffic-eng area 0
  rsvp
  interface pos 0/6/0/0
  bandwidth 100
commit
```

The following configuration example illustrates PCC configuration:

```
configure
  interface tunnel-te 10
  ipv4 unnumbered loopback 0
destination 1.2.3.4
  path-option 1 dynamic pce
mpls traffic-eng
  interface pos 0/6/0/0
  router id loopback 0
  router ospf 1
  router-id 192.168.25.66
  area 0
  interface pos 0/6/0/0
  interface loopback 0
  mpls traffic-eng router-id loopback 0
  mpls traffic-eng area 0
  rsvp
  interface pos 0/6/0/0
  bandwidth 100
commit
```
Configure Tunnels for Path Protection: Example

The path protection feature is configured only on the source router. The dynamic path option is a prerequisite to configure a path protection.

```
interface tunnel-te150
  ipv4 unnumbered Loopback150
  autoroute announce
destination 151.151.151.151
affinity 11 mask 11
path-protection
path-option 2 explicit name p2mp3-p2mp4-p2mp5
path-option 10 dynamic
```

Related Topics
- Enabling Path Protection for an Interface, page 78
- Assigning a Dynamic Path Option to a Tunnel, page 80
- Forcing a Manual Switchover on a Path-Protected Tunnel, page 81
- Configuring the Delay the Tunnel Takes Before Reoptimization, page 82
- Path Protection, page 18

Configure Automatic Bandwidth: Example

The following configuration example illustrates an automatic bandwidth configuration:

```
configure
interface tunnel-te6
  auto-bw
    bw-limit min 10000 max 500000
    overflow threshold 50 min 1000 limit 3
    adjustment-threshold 20 min 1000
    application 180
```

Related Topics
- Configuring the Collection Frequency, page 83
- Configuring the Automatic Bandwidth Functions, page 86
- MPLS-TE Automatic Bandwidth Overview, page 20

Configure the MPLS-TE Shared Risk Link Groups: Example

The following configuration example shows how to specify the SRLG value of each link that has a shared risk with another link:

```
config t
srlg
  interface POS0/4/0/0
    value 10
    value 11
  interface POS0/4/0/1
    value 10
```

Implementing MPLS Traffic Engineering
The following example shows the SRLG values configured on a specific link.

```
RP/0/RSP0/CPU0:router# show mpls traffic-eng topology brief
My_System_id: 100.0.0.2 (OSPF 0 area 0)
My_System_id: 0000.0000.0002.00 (IS-IS 1 level-1)
My_System_id: 0000.0000.0002.00 (IS-IS 1 level-2)
My_BC_Model_Type: RDM

Signalling error holdown: 10 sec Global Link Generation 389225
IGP Id: 0000.0000.0002.00, MPLS TE Id: 100.0.0.2 Router Node (IS-IS 1 level-1)
IGP Id: 0000.0000.0002.00, MPLS TE Id: 100.0.0.2 Router Node (IS-IS 1 level-2)

Link[1]:Broadcast, DR:0000.0000.0002.07, Nbr Node Id:21, gen:389193
  Frag Id:0, Intf Address:51.2.3.2, Intf Id:0  
  Nbr Intf Address:51.2.3.2, Nbr Intf Id:0
  TE Metric:10, IGP Metric:10, Attribute Flags:0x0
  Attribute Names:
    SRLGs: 1, 4, 5
  Switching Capability:, Encoding:
    BC Model ID:RDM
    Physical BW:1000000 (kbps), Max Reservable BW Global:10000 (kbps)
    Max Reservable BW Sub:10000 (kbps)

The following example shows the configured tunnels and associated SRLG values.

```
RP/0/RSP0/CPU0:router# show mpls traffic-eng tunnels
<snip>

Signalling Summary:
  LSP Tunnels Process: running
  RSVP Process: running
  Forwarding: enabled
  Periodic reoptimization: every 3600 seconds, next in 1363 seconds
  Periodic FRR Promotion: every 300 seconds, next in 181 seconds
  Auto-bw enabled tunnels: 0 (disabled)

Name: tunnel-te1 Destination: 100.0.0.3  
Status:  
  Admin: up Oper: up Path: valid Signalling: recovered
  path option 1, type explicit path123 (Basis for Setup, path weight 2)
  OSPF 0 area 0
  G-PID: 0x0800 (derived from egress interface properties)
  SRLGs excluded: 2,3,4,5  
  6,7,8,9
  Bandwidth Requested: 0 kbps CT0
<snip>
```

The following example shows all the interfaces associated with SRLG.

```
RP/0/RSP0/CPU0:router# show mpls traffic-eng topo srlg
My_System_id: 100.0.0.5 (OSPF 0 area 0)
My_System_id: 0000.0000.0005.00 (IS-IS 1 level-2)
My_System_id: 0000.0000.0005.00 (IS-IS ISIS-instance-123 level-2)

<table>
<thead>
<tr>
<th>SRLG</th>
<th>Interface Addr</th>
<th>TE Router ID</th>
<th>IGP Area ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>50.4.5.5</td>
<td>100.0.0.5</td>
<td>IS-IS ISIS-instance-123 level-2</td>
</tr>
<tr>
<td>11</td>
<td>50.2.3.3</td>
<td>100.0.0.3</td>
<td>IS-IS 1 level-2</td>
</tr>
<tr>
<td>12</td>
<td>50.2.3.3</td>
<td>100.0.0.3</td>
<td>IS-IS 1 level-2</td>
</tr>
<tr>
<td>30</td>
<td>50.4.5.5</td>
<td>100.0.0.5</td>
<td>IS-IS ISIS-instance-123 level-2</td>
</tr>
<tr>
<td>77</td>
<td>50.4.5.5</td>
<td>100.0.0.5</td>
<td>IS-IS ISIS-instance-123 level-2</td>
</tr>
<tr>
<td>88</td>
<td>50.4.5.5</td>
<td>100.0.0.5</td>
<td>IS-IS ISIS-instance-123 level-2</td>
</tr>
<tr>
<td>1500</td>
<td>50.4.5.5</td>
<td>100.0.0.5</td>
<td>IS-IS ISIS-instance-123 level-2</td>
</tr>
<tr>
<td>100000000</td>
<td>50.4.5.5</td>
<td>100.0.0.5</td>
<td>IS-IS ISIS-instance-123 level-2</td>
</tr>
</tbody>
</table>
```
The following example shows the NHOP and NNHOP backup tunnels with excluded SRLG values.

```
RP/0/RSP0/CPU0:router# show mpls traffic-eng topology path dest 100.0.0.5 exclude-srlg ipaddr
Path Setup to 100.0.0.2:
  bw 0 (CT0), min_bw 0, metric: 30
  setup_pri 7, hold_pri 7
  affinity_bits 0x0, affinity_mask 0xffff
  Exclude SRLG Intf Addr : 50.4.5.5
SRLGs Excluded : 10, 30, 1500, 1000000, 4294967290, 4294967295
Hop0:50.5.1.5
Hop1:50.5.1.1
Hop2:50.1.3.1
Hop3:50.1.3.3
Hop4:50.2.3.3
Hop5:50.2.3.2
Hop6:100.0.0.2
```

The following example shows an extract of explicit-path set to protect a specific interface.

```
RP/0/RSP0/CPU0:router#sh mpls traffic-eng topology path dest 10.0.0.5 explicit-path name name
Path Setup to 100.0.0.5:
  bw 0 (CT0), min_bw 9999, metric: 2
  setup_pri 7, hold_pri 7
  affinity_bits 0x0, affinity_mask 0xffff
SRLGs Excluded: 10, 30, 77, 88, 1500, 1000000, 4294967290, 4294967295
Hop0:50.3.4.3
Hop1:50.3.4.4
Hop2:50.4.5.4
Hop3:50.4.5.5
Hop4:100.0.0.5
```

**Related Topics**

- Configuring the SRLG Values of Each Link that has a Shared Risk with Another Link, page 89
- Creating an Explicit Path With Exclude SRLG, page 91
- Using Explicit Path With Exclude SRLG, page 92
- Creating a Link Protection on Backup Tunnel with SRLG Constraint, page 95
- Creating a Node Protection on Backup Tunnel with SRLG Constraint, page 98
- MPLS Traffic Engineering Shared Risk Link Groups, page 26

**Configure the MPLS-TE Auto-Tunnel Backup: Example**

The following example shows the auto-tunnel backup configuration for core or edge routers.

```
RP/0/RSP0/CPU0:router(config)#
mpls traffic-eng
  auto-tunnel backup
    tunnel-id min 60000 max 61000
    interface pos 0/1/0/0
      auto-tunnel backup
```
The following examples show the protection (NNHOP and SRLG) that was set on the auto-tunnel backup.

```
RP/0/RSP0/CPU0:router# show mpls traffic-eng tunnels 1
Signalling Summary:
  LSP Tunnels Process: running
  RSVP Process: running
  Forwarding: enabled
  Periodic reoptimization: every 3600 seconds, next in 2524 seconds
  Periodic FRR Promotion: every 300 seconds, next in 49 seconds
  Auto-bw enabled tunnels: 1
Name: tunnel-te1 Destination: 200.0.0.3 (auto backup)
Status:
  Admin: up Oper: up Path: valid Signalling: connected
  path option 10, type explicit (autob_nnhop_srlg_tunnel1) (Basis for Setup, path weight 11)
  path option 20, type explicit (autob_nnhop_tunnel1)
  G-PID: 0x0800 (derived from egress interface properties)
  Bandwidth Requested: 0 kbps CT0
  Creation Time: Fri Jul 10 01:53:25.581 PST (1h 25m 17s ago)
Config Parameters:
  Bandwidth: 0 kbps (CT0) Priority: 7 7 Affinity: 0x0/0xffff
  Metric Type: TE (default)
  AutoRoute: disabled LockDown: disabled Policy class: not set
  Forwarding-Adjacency: disabled
  Loadshare: 0 equal loadshares
  Auto-bw: disabled
  Fast Reroute: Disabled, Protection Desired: None
  Path Protection: Not Enabled
Auto Backup:
  Protected LSPs: 4
  Protected S2L Sharing Families: 0
  Protected S2Ls: 0
  Protected i/f: G10/1/0/0 Protected node: 20.0.0.2
  Protection: NNHOP+SRLG
  Unused removal timeout: not running
History:
  Tunnel has been up for: 00:00:08
  Current LSP:
    Uptime: 00:00:08
    Prior LSP:
      ID: path option 1 [545]
      Removal Trigger: configuration changed
Path info (OSPF 0 area 0):
  Hop0: 10.0.0.2
  Hop1: 100.0.0.2
  Hop2: 100.0.0.3
  Hop3: 200.0.0.3
```

The following example shows automatically created path options for this backup auto-tunnel.

```
RP/0/RSP0/CPU0:router# show mpls traffic-eng tunnels 1 detail
Signalling Summary:
  LSP Tunnels Process: running
  RSVP Process: running
  Forwarding: enabled
  Periodic reoptimization: every 3600 seconds, next in 2524 seconds
  Periodic FRR Promotion: every 300 seconds, next in 49 seconds
  Auto-bw enabled tunnels: 1
Name: tunnel-te1 Destination: 200.0.0.3 (auto backup)
Status:
  Admin: up Oper: up Path: valid Signalling: connected
  path option 10, type explicit (autob_nnhop_srlg_tunnel1) (Basis for Setup, path weight 11)
```

The following example shows automatically created path options for this backup auto-tunnel.
Configure the MPLS-TE Auto-Tunnel Backup: Example

This example shows the automatically created backup tunnels.

```
RP/0/RSP0/CPU0# show mpls traffic-eng tunnels brief
TUNNEL NAME    DESTINATION    STATUS    STATE
--------------- --------------- -------------
tunnel-te0      200.0.0.3     up            up
  tunnel-te1     200.0.0.3     up            up
  tunnel-te2     200.0.0.3     up            up
  tunnel-te50    200.0.0.3     up            up
*tunnel-te60    200.0.0.3     up            up
*tunnel-te70    200.0.0.3     up            up
*tunnel-te80    200.0.0.3     up            up

RP/0/RSP0/CPU0# show mpls traffic-eng tunnels tabular

<table>
<thead>
<tr>
<th>Tunnel Name</th>
<th>LSP</th>
<th>Destination</th>
<th>Source</th>
<th>FRR</th>
<th>LSP</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ID</td>
<td>Address</td>
<td>Address</td>
<td>State</td>
<td>State</td>
<td>Role</td>
</tr>
<tr>
<td>---------------</td>
<td>------</td>
<td>-------------</td>
<td>--------</td>
<td>-----</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td>tunnel-te0</td>
<td>549</td>
<td>200.0.0.3</td>
<td>200.0.0.1</td>
<td>up</td>
<td>Inact</td>
<td>Head</td>
</tr>
<tr>
<td>tunnel-te1</td>
<td>546</td>
<td>200.0.0.3</td>
<td>200.0.0.1</td>
<td>up</td>
<td>Inact</td>
<td>Head</td>
</tr>
<tr>
<td>tunnel-te2</td>
<td>6</td>
<td>200.0.0.3</td>
<td>200.0.0.1</td>
<td>up</td>
<td>Inact</td>
<td>Head</td>
</tr>
<tr>
<td>tunnel-te50</td>
<td>6</td>
<td>200.0.0.3</td>
<td>200.0.0.1</td>
<td>up</td>
<td>Inact</td>
<td>Head</td>
</tr>
<tr>
<td>tunnel-te60</td>
<td>4</td>
<td>200.0.0.3</td>
<td>200.0.0.1</td>
<td>up</td>
<td>Active</td>
<td>Head</td>
</tr>
<tr>
<td>tunnel-te70</td>
<td>4</td>
<td>200.0.0.3</td>
<td>200.0.0.1</td>
<td>up</td>
<td>Active</td>
<td>Head</td>
</tr>
<tr>
<td>tunnel-te80</td>
<td>3</td>
<td>200.0.0.3</td>
<td>200.0.0.1</td>
<td>up</td>
<td>Active</td>
<td>Head</td>
</tr>
</tbody>
</table>
```
Configure Point-to-Multipoint TE: Examples

These configuration examples show how to configure Point-to-Multipoint TE:

P2MP Topology Scenario: Example

This section describes a typical scenario of point-to-multipoint traffic engineering topology. This figure illustrates the P2MP topology.

![Figure 10: P2MP Topology](image)

This head router describes the configuration at head node. This router does the imposition of MPLS at head node.

```plaintext
interface tunnel-mte1
  ipv4 unnumbered Loopback0
destination 1.1.1.1
  path-option 1 explicit name path-to-tail1
  !destination 2.2.2.2
  path-option 1 explicit name path-to-tail2
  !fast-reroute
mpls traffic-eng
  interface GigabitEthernet0/1/3/0
  !interface GigabitEthernet0/1/3/7

multicast-routing
  address-family ipv4
    nsf
    interface all enable
  !address-family ipv6
    nsf
    interface all enable
```

Related Topics

- Enabling an AutoTunnel Backup, page 43
- Removing an AutoTunnel Backup, page 44
- Establishing MPLS Backup AutoTunnels to Protect Fast Reroutable TE LSPs, page 46
- Establishing Next-Hop Tunnels with Link Protection, page 47
- Backup AutoTunnels, page 4
This mid router describes the configuration at mid node. This router performs the role of MPLS label replication at mid node.

```conf
! router igmp
vrf default
  interface tunnel-mte1
    static-group 232.0.0.1 192.168.10.1
!
```

This tail router describes the configuration at tail node. This router performs the role of MPLS disposition at tail node.

```conf
! mpls traffic-eng
  interface POS0/2/0/0
  ! interface POS0/2/0/1
    backup-path tunnel-te 1000
  ! interface TenGigE0/3/0/3
  ! interface GigabitEthernet0/2/5/0
  !
! multicast-routing
  address-family ipv4
    interface all enable
    ! core-tree-protocol rsvp-te group-list lsm
      static-rpf 192.168.10.1 32 mpls 5.5.5.5
    !
!
```

This configuration describes the Fast Reroute configuration in the MPLS network.

```conf
explicit-path name backup-path-to-tail1
  index 1 next-address strict 198.1.1.2
  index 2 next-address strick 198.1.2.2
  !
  interface tunnel-te1000 <<< backup p2p tunnel
    ipv4 unnumbered Loopback0
    destination 140.140.140.140
    path-option 1 explicit name backup-path-to-tail1
  ! mpls traffic-eng
  interface POS0/2/0/0
  ! interface POS0/2/0/1
    backup-path tunnel-te 1000
  ! interface TenGigE0/5/0/4
  !
```

**Configure Point-to-Multipoint for the Source: Example**

At the source, multicast routing must be enabled on both the tunnel-mte interface and customer-facing interface. Then, the static-group must be configured on the tunnel-mte interface to forward specified multicast traffic over P2MP LSP.
The multicast group address, which is in Source-Specific Multicast (SSM) address range (ff35::/16), must be used on the static-group configuration because Cisco IOS XR software supports only SSM for Label Switch Multicast (LSM). Additionally, the customer-facing interface must have an IPv6 address.

```
multicast-routing
   address-family ipv6
   interface tunnel-mte 1
      enable
      !
      interface GigabitEthernet0/2/0/3
         enable
      !
      !
      router mld
      vrf default
      interface tunnel-mte 1
         static-group ff35::1 2000::1 3eFF::A
      !
      !
   interface tunnel-mte 1
      ipv4 unnumbered Loopback0
      destination 3.3.3.3
      path-option 1 dynamic
      destination 4.4.4.4
      path-option 1 dynamic
      !
      !
```

**Related Topics**

- Point-to-Multipoint Traffic-Engineering Overview, page 22
- Point-to-Multipoint RSVP-TE, page 24

**Configure the Point-to-Multipoint Tunnel: Example**

There is no difference between logging events at the tunnel level for both P2P and P2MP. The P2MP tunnel reoptimizes only at the per tunnel level.

```
interface tunnel-mte1
   ipv4 unnumbered Loopback0
   destination 60.60.60.60
   logging events lsp-status reroute
   path-option 10 explicit name toR6_via_R2andR3
      !
   logging events lsp-status reoptimize
   logging events lsp-status state
   logging events lsp-status reroute
   fast-reroute
   record-route
      !
   explicit-path name PATH7
      index 1 next-address strict ipv4 unicast 192.168.7.2
      index 2 next-address strict ipv4 unicast 192.168.7.1
      index 3 next-address strict ipv4 unicast 192.168.16.1
      index 4 next-address strict ipv4 unicast 192.168.16.2
      !
```
Disable a Destination: Example

From the tunnel-mte interface, you can disable the destination.

```
interface tunnel-mte101
  ipv4 unnumbered Loopback0
  destination 150.150.150.150
disable
  path-option 10 dynamic
!
```

Configure the Point-to-Multipoint Solution: Example

Requirements for MPLS-TE Configuration

Before the Point-to-Multipoint (P2MP) tunnel is defined, these MPLS-TE requirements must be configured:

- Multiprotocol Label Switching traffic engineering (MPLS-TE)
- Resource ReSerVation Protocol (RSVP)
- Open Shortest Path First (OSPF)

This example shows the entire P2MP solution:

- Source is the location where the P2MP-TE tunnel interface is created.
- Tunnel contains multiple destinations. For example, the P2MP-TE tunnel is configured with two leaf node destinations by using the dynamic and explicit path options.
- Fast-Reroute (FRR) is specified on the P2MP tunnel.
- All regular TE tunnel options such as affinity or bandwidth are configured.
- Static mapping of the group address to the P2MP tunnel is done in IGMP.
- The P2MP-TE midpoint configuration requires only TE and Interior Gateway Protocol (IGP) information.
- The P2MP-TE receiver configuration requires a static group and RPF map.

```
! explicit-path name g2-r2-r1
  index 1 next-address strict ipv4 unicast 10.2.15.1
!
! explicit-path name g2-r2-r3
  index 1 next-address strict ipv4 unicast 10.2.25.1
```
Implementing MPLS Traffic Engineering

Configure Point-to-Multipoint TE: Examples

```
! index 2 next-address strict ipv4 unicast 10.2.23.2
! explicit-path name g2-r2-r4
index 1 next-address strict ipv4 unicast 10.2.25.1
index 2 next-address strict ipv4 unicast 10.2.24.2
!
ipv4 access-list ssm
10 permit ipv4 232.1.0.0/16 any
20 permit ipv4 232.3.0.0/16 any
30 permit ipv4 232.4.0.0/16 any
!
ipv4 access-list ssm-test
10 permit ipv4 235.0.0.0/8 any
!
interface Loopback0
ipv4 address 192.168.1.2 255.255.255.255
!
interface tunnel-mte221
ipv4 unnumbered Loopback0
destination 192.168.1.1
path-option 1 dynamic
!
destination 192.168.1.3
path-option 1 dynamic
!
destination 192.168.1.4
path-option 1 dynamic
!
interface tunnel-mte222
ipv4 unnumbered Loopback0
destination 192.168.1.1
path-option 1 explicit name g2-r2-r1
!
destination 192.168.1.3
path-option 1 explicit name g2-r2-r3
!
destination 192.168.1.4
path-option 1 explicit name g2-r2-r4
!
signalled-bandwidth 1000
!
interface MgmtEth0/RP0/CPU0/0
ipv4 address 172.20.163.12 255.255.255.128
!
interface MgmtEth0/RP1/CPU0/0
shutdown
!
interface GigabitEthernet0/0/0/0
ipv4 address 172.2.1.2 255.255.255.0
load-interval 30
!
interface GigabitEthernet0/0/0/0/1
ipv4 address 10.1.15.2 255.255.255.0
!
interface GigabitEthernet0/0/0/1.2
ipv4 address 10.2.15.2 255.255.255.0
dot1q vlan 2
!
interface GigabitEthernet0/0/0/2
ipv4 address 10.1.25.2 255.255.255.0
!
interface GigabitEthernet0/0/0/0/2
ipv4 address 10.2.25.2 255.255.255.0
dot1q vlan 2
!
interface GigabitEthernet0/0/0/3
shutdown
!
interface GigabitEthernet0/0/0/4
shutdown
!
interface GigabitEthernet0/0/0/5
```
Implementing MPLS Traffic Engineering

Configure Point-to-Multipoint TE: Examples

```
shutdown
!
interface GigabitEthernet0/0/0/6
shutdown
!
interface GigabitEthernet0/0/0/7
shutdown
!
router static
  address-family ipv4 unicast
    0.0.0.0/0 1.56.0.1
    0.0.0.0/0 172.20.163.1
!
router ospf 100
  nsr
    router-id Loopback0
  area 0
    mpls traffic-eng
    interface Loopback0
      !
      interface GigabitEthernet0/0/0/0
      !
      interface GigabitEthernet0/0/0/1
      !
      interface GigabitEthernet0/0/0/2
      !
      interface GigabitEthernet0/0/0/1.2
      !
      interface GigabitEthernet0/0/0/2.2
      !
      mpls traffic-eng router-id Loopback0
      !
      mpls oam
      !
      rsvp
        interface GigabitEthernet0/0/0/0
          bandwidth 20000
        !
        interface GigabitEthernet0/0/0/1
          bandwidth 20000
        !
        interface GigabitEthernet0/0/0/2
          bandwidth 20000
        !
        interface GigabitEthernet0/0/0/1.2
          bandwidth 20000
        !
        interface GigabitEthernet0/0/0/2.2
          bandwidth 20000
        !
        mpls traffic-eng
        interface GigabitEthernet0/0/0/0
        !
        interface GigabitEthernet0/0/0/1
        !
        interface GigabitEthernet0/0/0/2
        !
        interface GigabitEthernet0/0/0/1.2
        !
        interface GigabitEthernet0/0/0/2.2
        !
        mpls ldp
        router-id 192.168.1.2
        nsr
        graceful-restart
        interface GigabitEthernet0/0/0/0
        !
        interface GigabitEthernet0/0/0/1
        !
```
interface GigabitEthernet0/0/0/1.2
!
interface GigabitEthernet0/0/0/2
!
interface GigabitEthernet0/0/0/2.2
!
multicast-routing
address-family ipv4
core-tree-protocol RSVP-TE
ssm range ssm
static-rpf 172.1.1.1 32 mpls 192.168.1.1
static-rpf 172.3.1.1 32 mpls 192.168.1.3
static-rpf 172.4.1.1 32 mpls 192.168.1.4
interface all enable
!
router igmp
!
interface tunnel-mte221
  static-group 232.2.2.1 172.2.1.1
!
interface tunnel-mte222
  static-group 232.2.2.2 172.2.1.1
!
interface GigabitEthernet0/0/0/0
  static-group 232.1.2.1 172.1.1.1
  static-group 232.1.2.2 172.1.1.1
  static-group 232.3.2.1 172.3.1.1
  static-group 232.3.2.2 172.3.1.1
  static-group 232.4.2.1 172.4.1.1
  static-group 232.4.2.2 172.4.1.1
!
end

Related Topics

- Point-to-Multipoint Traffic-Engineering Overview, page 22
- Point-to-Multipoint RSVP-TE, page 24
- Path Option for Point-to-Multipoint RSVP-TE, page 25
- Point-to-Multipoint Traffic-Engineering Overview, page 22

Additional References

For additional information related to implementing MPLS-TE, refer to the following references:

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPLS-TE commands</td>
<td>MPLS Traffic Engineering Commands on Cisco ASR 9000 Series Router module in</td>
</tr>
<tr>
<td></td>
<td>Cisco ASR 9000 Series Aggregation Services Router MPLS Command Reference</td>
</tr>
<tr>
<td>Getting started material</td>
<td>Cisco ASR 9000 Series Aggregation Services Router Getting Started Guide</td>
</tr>
</tbody>
</table>
## Standards

<table>
<thead>
<tr>
<th>Standards</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.</td>
<td>—</td>
</tr>
</tbody>
</table>

## MIBs

<table>
<thead>
<tr>
<th>MIBs</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>—</td>
<td>To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: <a href="http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml">http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml</a></td>
</tr>
</tbody>
</table>

## RFCs

<table>
<thead>
<tr>
<th>RFCs</th>
<th>Title</th>
</tr>
</thead>
</table>

## Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.</td>
<td><a href="http://www.cisco.com/techsupport">http://www.cisco.com/techsupport</a></td>
</tr>
</tbody>
</table>