# terminating-pe tie-breaker

To negotiate the behavior mode (either active or passive) for a terminating provider edge (TPE) router, use the **terminating-pe tie-breaker** command in Layer 2 pseudowire routing configuration mode. To remove the TPE tie breaker identification, use the **no** form of this command.

**terminating-pe tie-breaker**

**no terminating-pe tie-breaker**

## Syntax Description

This command has no arguments or keywords.

## Command Default

A behavior mode is not specified for the TPE.

## Command Modes

Layer 2 pseudowire routing (config-l2_pw_rtg)

## Command History

| Release | Modification |
|---|---|
| 15.1(1)S | This command was introduced. |

## Usage Guidelines

The **terminating-pe** command is used in Layer 2 pseudowire routing configuration mode. To enter Layer 2 pseudowire routing configuration mode, use the **l2 pseudowire routing** command.

### Active and Passive PEs in an L2VPN VPLS Inter-AS Option B Configuration

A TPE terminates a multisegment pseudowire. By default, the TPEs on both ends of a multisegmented pseudowire are in active mode. The L2VPN VPLS Inter-AS Option B feature requires that one of the TPEs be in passive mode. The system determines which PE is the passive TPE based on a comparison of the Target Attachment Individual Identifier (TAII) received from Border Gateway Protocol (BGP) and the Source Attachment Individual Identifier (SAII) of the local router. The TPE with the numerically higher identifier assumes the active role.

When you are configuring the PEs for the L2VPN VPLS Inter-AS Option B feature, use the **terminating-pe tie-breaker** command to negotiate the mode of the TPE. Then use the **mpls ldp discovery targeted-hello accept** command to ensure that a passive TPE can accept Label Distribution Protocol (LDP) sessions from the LDP peers.

## Examples

In the following example, the **terminating-pe** command has been used to configure the TPE to negotiate an active or passive role:

```
Router>
Router# enable
Router(config)# configure terminal
Router(config)# l2 pseudowire routing
Router(config-l2_pw_rtg)# terminating-pe tie-breaker
Router(config-l2_pw_rtg)# end
```

## Related Commands

| Command | Description |
|---|---|
| **l2 pseudowire routing** | Enables Layer 2 pseudowire routing and enters Layer 2 pseudowire routing configuration mode. |
| **mpls ldp discovery** | Configures the interval between transmission of consecutive LDP discovery hello messages, or the hold time for a discovered LDP neighbor, or the neighbors from which requests for targeted hello messages may be honored. |
| **show xconnect** | Displays information about xconnect attachment circuits and pseudowires. |

# tag-control-protocol vsi

**Note** Effective with Cisco IOS Release 12.4(20)T, the **tag-control-protocol vsi** command is not available in Cisco IOS software.

To configure the use of Virtual Switch Interface (VSI) on a particular master control port, use the **tag-control-protocol vsi** command in interface configuration mode. To disable VSI, use the **no** form of this command.

**tag-control-protocol vsi** [**base-vc** *vpi vci*] [**delay** *seconds*] [**id** *controller-id*] [**keepalive** *timeout*] [**nak** [**basic** | **extended**]] [**retry** *timeout-count*] [**slaves** *slave-count*]

**no tag-control-protocol vsi** [**base-vc** *vpi vci*] [**delay** *seconds*] [**id** *controller-id*] [**keepalive** *timeout*] [**nak** [**basic** | **extended**]] [**retry** *timeout-count*] [**slaves** *slave-count*]

## Syntax Description

| | |
|---|---|
| **base-vc** *vpi vci* | (Optional) Determines the VPI/VCI value for the channel to the first slave. The default is 0/40.<br>Together with the slave value, this value determines the VPI/VCI values for the channels to all of the slaves, which are as follows:<br>• *vpi/vci*<br>• *vpi/vci*+1, and so on<br>• *vpi/vci*+*slave-count*-1 |
| **delay** *seconds* | (Optional) Specifies the delay time to start a new VSI session after the system comes up or after you enter the command. If a VSI session is already running, the **delay** keyword has no effect for the current session. The delay is implemented when a new VSI session starts. The default is 0. The valid range of values is 0 to 300. |
| **id** *controller-id* | (Optional) Determines the value of the controller-id field present in the header of each VSI message. The default is 1. |
| **keepalive** *timeout* | (Optional) Determines the value of the keepalive timer (in seconds). Make sure that the keepalive timer value is greater than the value of the retry timer times the retry timer +1. The default is 15 seconds. |

| | |
|---|---|
| **nak** [**basic** \| **extended**] | (Optional) Allows the label switch controller (LSC) to request extended negative acknowledgment (NAK) responses from the VSI slave. The extended NAK response indicates a dangling connection on the VSI slave. If the slave sends an extended NAK response code, the LSC sends a delete connection command that enables the VSI slave to delete the dangling connection.<br><br>Use the **basic** keyword to specify the NAK 11 and NAK 12 response codes from the VSI. If you use the **nak basic** keywords, support for extended NAK is not enabled on the LSC. The interface configuration does not indicate that basic NAK support is enabled. The output of the **show controller vsi session** command does not indicate that basic NAK support is enabled.<br><br>Use the **extended** keyword to specify extended NAK codes 51 - 54 from the VSI, which are supported in VSI protocol version 2.4. If you use the **nak extended** keywords, support for extended NAK is enabled on the LSC. The interface configuration indicates that extended NAK support is enabled. The output of the **show controller vsi session** command also indicates that extended NAK support is enabled.<br><br>**Note** Use the **nak extended** keyword only if all VSI slaves support extended NAK codes. |
| **retry** *timeout-count* | (Optional) Determines the value of the message retry timer (in seconds) and the maximum number of retries. The default is 8 seconds and 10 retries. |
| **slaves** *slave-count* | (Optional) Determines the number of slaves reachable through this master control port. The default is 14 (suitable for the Cisco BPX switch). |

## Defaults

VSI is disabled.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 12.0(5)T | This command was introduced. |
| 12.2(15)T | The **delay** keyword was added. |
| 12.3(2)T | The **nak** keyword was added. |
| 12.4(20)T | This command was removed. |

## Usage Guidelines

- The command is only available on interfaces that can serve as a VSI master control port. Cisco recommends that all options to the **tag-control-protocol vsi** command be entered at the same time.
- After VSI is active on the control interface (through the earlier issuance of a **tag-control-protocol vsi** command), reentering the command may cause all associated XTagATM interfaces to shut down and restart. In particular, if you reenter the **tag-control-protocol vsi** command with any of the following options, the VSI shuts down and reactivates on the control interface:
  - **id**
  - **base-vc**

– **slaves**

The VSI remains continuously active (that is, the VSI does not shut down and then reactivate) if you reenter the **tag-control-protocol vsi** command with only one or both of the following options:

– **keepalive**

– **retry**

– **delay**

In either case, if you reenter the **tag-control-protocol vsi** command, this causes the specified options to take on the newly specified values; the other options retain their previous values. To restore default values to all the options, enter the **no tag-control-protocol** command, followed by the **tag-control-protocol vsi** command.

## Examples

The following example shows how to configure the VSI driver on the control interface:

```
Router(config)# interface atm 0/0
Router(config-if)# tag-control-protocol vsi base-vc 0 51
```

The following example enables extended NAK support:

```
Router(config-if)# tag-control-protocol vsi nak extended
```

The following example shows that extended NAK support is enabled, as shown by the bold output:

```
Router# show running-config interface atm0/0

Building configuration...
Current configuration : 113 bytes
interface ATM0/0
 no ip address
 shutdown
 label-control-protocol vsi nak extended
 no atm ilmi-keepalive
end
```

The **show controllers vsi session** command also indicates that extended NAK support is enabled, as shown by the bold output:

```
Router# show controllers vsi session

Interface     Session  VCD    VPI/VCI     Switch/Slave Ids   Session State
ATM0/0        0        1      0/40        0/0                UNKNOWN
ATM0/0        1        2      0/41        0/0                UNKNOWN
ATM0/0        2        3      0/42        0/0                UNKNOWN
ATM0/0        3        4      0/43        0/0                UNKNOWN
ATM0/0        4        5      0/44        0/0                UNKNOWN
ATM0/0        5        6      0/45        0/0                UNKNOWN
ATM0/0        6        7      0/46        0/0                UNKNOWN
ATM0/0        7        8      0/47        0/0                UNKNOWN
ATM0/0        8        9      0/48        0/0                UNKNOWN
ATM0/0        9        10     0/49        0/0                UNKNOWN
ATM0/0        10       11     0/50        0/0                UNKNOWN
ATM0/0        11       12     0/51        0/0                UNKNOWN
ATM0/0        12       13     0/52        0/0                UNKNOWN
ATM0/0        13       14     0/53        0/0                UNKNOWN
Extended NAK support is enabled on LSC
```

Table 173 describes the significant fields shown in the display.

*Table 173 show controllers vsi session Field Descriptions*

| Field | Description |
|---|---|
| Interface | Control interface name. |
| Session | Session number (from 0 to <*n*–1>), where *n* is the number of sessions on the control interface. |
| VCD | Virtual circuit descriptor (virtual circuit number). Identifies the VC carrying the VSI protocol between the master and the slave for this session. |
| VPI/VCI | Virtual path identifier or virtual channel identifier (for the VC used for this session). |
| Switch/Slave Ids | Switch and slave identifiers supplied by the switch. |
| Session State | Indicates the status of the session between the master and the slave.<br>• ESTABLISHED is the fully operational steady state.<br>• UNKNOWN indicates that the slave is not responding.<br>Other possible states include the following:<br>• CONFIGURING<br>• RESYNC-STARTING<br>• RESYNC-UNDERWAY<br>• RESYNC-ENDING<br>• DISCOVERY<br>• SHUTDOWN-STARTING<br>• SHUTDOWN-ENDING<br>• INACTIVE |

# tlv

To specify the pseudowire type-length-value (TLV) parameters, use the **tlv** command, in virtual forwarding interface (VFI) neighbor interface configuration mode or pseudowire TLV template configuration mode. To remove the TLV parameters, use the **no** form of this command.

**tlv** [*type-name*] *type-value length* [**dec** | **hexstr** | **str**] *value*

**no tlv** [*type-name*] *type-value length* [**dec** | **hexstr** | **str**] *value*

## Syntax Description

| | |
|---|---|
| *type-name* | The name of the TLV. |
| *type-value* | A number designating the type of TLV. Valid values are from 1 to 40. |
| *length* | The TLV length. Valid values are from 1 to 255. |
| **dec** | The TLV value in decimal. |
| **hexstr** | The TLV value in hex string. |
| **str** | The TLV value in string. |
| *value* | The TLV value. |

## Command Default

No defaults

## Command Modes

VFI neighbor interface configuration (config-vfi-neighbor-interface)

PseudowireTLV template configuration (config-pw-tlv-template)

## Command History

| Release | Modification |
|---|---|
| 15.1(1)SA | This command was introduced. |
| 15.1(3)S | This command was integrated. |

## Usage Guidelines

Use this command with the MPLS-TP feature set.

## Examples

The following example specifies TLV values:

```
l2 vfi atom point-to-point  (static-dynamic MSPW)
neighbor 116.116.116.116 4294967295 pw-class dypw     (dynamic)
neighbor 111.111.111.111 123 pw-class stpw            (static)
  mpls label 101 201
  mpls control-word
   local interface 4
     tlv mtu 1 4  1500
     tlv descr 3 6 str abcd
     tlv descr C 4 hexstr 0505
```

## Related Commands

| Command | Description |
|---|---|
| **pseudowire-tlv template** | Creates a template of TLV parameters to use in an MPLS-TP configuration. |

# tlv template

To use the pseudowire type-length-value (TLV) parameters created with the **pseudowire-tlv template** command, use the **tlv template** command in VFI neighbor interface configuration mode. To remove the TLV template, use the **no** form of this command.

**tlv template** *template-name*

**no tlv template** *template-name*

## Syntax Description

| | |
|---|---|
| *template-name* | The name of the TLV tempate you created with the **pseudowire-tlv template** command. |

## Command Default

No template is used.

## Command Modes

VFI neighbor interface configuration (config-vfi-neighbor-interface)

## Command History

| Release | Modification |
|---|---|
| 15.1(1)SA | This command was introduced. |
| 15.1(3)S | This command was integrated. |

## Usage Guidelines

Make sure that you create the template with the **pseudowire-tlv template** command before specifying the template as part of the local interface configuration.

## Examples

The following example uses the pseudowrire TLV template called net:

```
Router(config-vfi-neighbor-interface)# tlv template net
```

## Related Commands

| Command | Description |
|---|---|
| **pseudowire-tlv template** | Creates a template of TLV parameters to use in an MPLS-TP configuration. |

# trace mpls

To discover Multiprotocol Label Switching (MPLS) label switched path (LSP) routes that packets actually take when traveling to their destinations, use the **trace mpls** command in privileged EXEC mode.

**trace mpls**
{**ipv4** *destination-address/destination-mask-length*
| **traffic-eng Tunnel** *tunnel-number*
| **pseudowire** *destination-address vc-id* **segment** *segment-number* [*segment number*]}
[**timeout** *seconds*]
[**destination** *address-start* [*address-end* | *address-increment*]]
[**revision** {**1** | **2** | **3** | **4**}]
[**source** *source-address*]
[**exp** *exp-bits*]
[**ttl** *maximum-time-to-live*]
[**reply** {**dscp** *dscp-bits* | **mode** *reply-mode* {**ipv4** | **no-reply** | **router-alert**} | **pad-tlv**}]
[**force-explicit-null**]
[**output interface** *tx-interface* [**nexthop** *ip-address*]]
[**flags fec**]
[**revision** *tlv-revision-number*]

## Syntax Description

| | |
|---|---|
| **ipv4** | Specifies the destination type as a Label Distribution Protocol (LDP) IPv4 address. |
| *destination-address* | Address prefix of the target to be tested. |
| */destination-mask-length* | Number of bits in the network mask of the target address. The slash is required. |
| **traffic-eng Tunnel** *tunnel-number* | Specifies the destination type as an MPLS traffic engineering (TE) tunnel. |
| **pseudowire** | Specifies the destination type as an Any Transport over MPLS (AToM) virtual circuit (VC). |
| *ipv4-address* | IPv4 address of the AToM VC to be tested. |
| *vc-id* | Specifies the VC identifier of the AToM VC to be tested. |
| **segment** | Specifies a segment of a multisegment pseudowire. |
| *segment-number* | A specific segment of the multisegment pseudowire or a range of segments, indicated by two segment numbers. |
| **timeout** *seconds* | (Optional) Specifies the timeout interval in seconds. The range is from 0 to 3600. The default is 2 seconds. |
| **destination** | (Optional) Specifies a network 127 address. |
| *address-start* | (Optional) The beginning network 127 address. |
| *address-end* | (Optional) The ending network 127 address. |
| *address-increment* | (Optional) Number by which to increment the network 127 address. |

| | |
|---|---|
| **revision** {**1** \| **2** \| **3** \| **4**} | (Optional) Selects the type, length, values (TLVs) version of the implementation. Use the **revision 4** default unless attempting to interoperate with devices running Cisco IOS Release 12.0(27)S1 or 12.0(27)S2. If you do not select a **revision** keyword, the software uses the latest version.<br><br>See Table 174 in the"Revision Keyword Usage" section of the "Usage Guidelines" section for information on when to select the **1**, **2**, **3**, and **4** keywords. |
| **source** *source-address* | (Optional) Specifies the source address or name. The default address is loopback0. This address is used as the destination address in the MPLS echo response. |
| **exp** *exp-bits* | (Optional) Specifies the MPLS experimental field value in the MPLS header for an MPLS echo reply. Valid values are from 0 to 7. Default is 0. |
| **ttl** *maximum-time-to-live* | (Optional) Specifies a maximum hop count. Default is 30. |
| **reply dscp** *dscp-bits* | (Optional) Provides the capability to request a specific class of service (CoS) in an echo reply by providing a differentiated services code point (DSCP) value.<br><br>The echo reply is returned with the IP header ToS byte set to the value specified in the **reply dscp** keyword. |
| **reply mode** *reply-mode* | (Optional) Specifies the reply mode for the echo request packet.<br><br>The *reply-mode* is one of the following:<br><br>**ipv4**—Reply with an IPv4 User Datagram Protocol (UDP) packet (default).<br><br>**no-reply**—Do not send an echo request packet in response.<br><br>**router-alert**—Reply with an IPv4 UDP packet with router alert. |
| **reply pad-tlv** | (Optional) Tests the ability of the sender of an echo reply to support the copy pad TLV to echo reply. |
| **force-explicit-null** | (Optional) Forces an explicit null label to be added to the MPLS label stack even though the label was unsolicited. |
| **output interface** *tx-interface* | (Optional) Specifies the output interface for echo requests. |
| **nexthop** *ip-address* | (Optional) Causes packets to go through the specified next-hop address. |
| **flags fec** | (Optional) Requests that target Forwarding Equivalence Class (FEC) stack validation be done at the egress router. A downstream map TLV containing the correct received labels must be present in the echo request for target FEC stack checking to be performed.<br><br>Be sure to use this keyword in conjunction with the **ttl** keyword. |
| **revision** *tlv-revision-number* | (Optional) Cisco TLV revision number. |

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| 12.0(27)S | This command was introduced. |
| 12.2(18)SXE | The **reply dscp** and **reply pad-tlv** keywords were added. |
| 12.4(6)T | The following keywords were added: **force-explicit-null**, **output interface**, **flags fec**, and **revision**. |
| 12.2(28)SB | This command was integrated into Cisco IOS Release 12.2(28)SB and implemented on the Cisco 10000 series routers. |
| 12.0(32)SY | This command was integrated into Cisco IOS Release 12.0(32)SY. |
| 12.4(11)T | This command was integrated into Cisco IOS Release 12.4(11)T. |
| 12.2(31)SB2 | This command was integrated into Cisco IOS Release 12.2(31)SB2. The **nexthop** keyword was added. |
| 12.2(33)SRB | This command was integrated into Cisco IOS Release 12.2(33)SRB. |
| 12.2(33)SXH | This command was integrated into Cisco IOS Release 12.2(33)SXH. |
| 12.0(33)S | This command was integrated into Cisco IOS Release 12.0(33)S. |
| 12.4(20T | This command was integrated into Cisco IOS Release 12.4(20)T. |
| 12.2(33)SXI | This command was integrated into Cisco IOS Release 12.2(33)SXI. |
| Cisco IOS XE Release 2.3 | The **segment** keyword was added. |
| 12.2(33)SRE | This command was modified. Restrictions were added to the **pseudowire** keyword. |

## Usage Guidelines

Use the **trace mpls** command to validate, test, or troubleshoot IPv4 LDP LSPs and IPv4 Resource Reservation Protocol (RSVP) TE tunnels.

### UDP Destination Address Usage

The destination address is a valid 127/8 address. You can specify a single address or a range of numbers from 0.0.0 to *x.y.z*, where *x*, *y*, and *z* are numbers from 0 to 255 and correspond to the 127.*x.y.z* destination address.

The MPLS echo request destination address in the UDP packet is not used to forward the MPLS packet to the destination router. The label stack that is used to forward the echo request routes the MPLS packet to the destination router. The 127/8 address guarantees that the packets are routed to the localhost (the default loopback address of the router processing the address) if the UDP packet destination address is used for forwarding.

In addition, the destination address is used to adjust load balancing when the destination address of the IP payload is used for load balancing.

### Time-to-Live Keyword Usage

The time-to-live value indicates the maximum number of hops a packet should take to reach its destination. The value in the TTL field in a packet is decremented by 1 each time the packet travels through a router.

For MPLS LSP ping, the TTL is a value after which the packet is discarded and an MPLS echo reply is sent back to the originating router.

For MPLS Multipath LSP Traceroute, the TTL is a maximum time-to-live value and is used to discover the number of downstream hops to the destination router. MPLS LSP Traceroute incrementally increases the TTL value in its MPLS echo requests (TTL = 1, 2, 3, 4, ...) to accomplish this.

### Pseudowire Usage

The following keywords are not available with the **trace mpls pseudowire** command:

- **flags**
- **force-explicit-null**
- **output**
- **revision**
- **ttl**

### Revision Keyword Usage

The **revision** keyword allows you to issue a **trace mpls ipv4** or **trace mpls traffic-eng** command based on the format of the TLV. Table 174 lists the revision option and usage guidelines for each option.

*Table 174 Revision Options and Option Usage Guidelines*

| Revision Option | Option Usage Guidelines |
|---|---|
| 1[1] | Not supported in Cisco IOS Release 12.4(11)T or later releases.<br>Version 1 (draft-ietf-mpls-ping-03)<br>For a device running Cisco IOS Release 12.0(27)S3 or a later release, you must use the **revision 1** keyword when you send LSP ping or LSP traceroute commands to devices running Cisco IOS Release 12.0(27)S1 or 12.0(27)S2. |
| 2 | Version 2 functionality was replaced by Version 3 functionality before any images were shipped. |
| 3 | Version 3 (draft-ietf-mpls-ping-03).<br>• For a device implementing Version 3 (Cisco IOS Release 12.0(27)S3 or a later release), you must use the **revision 1** keyword when you send the LSP **ping** or LSP **traceroute** command to a device implementing Version 1 (that is, either Cisco IOS Release 12.0(27)S1 or Release 12.0(27)S2).<br>• A **ping mpls pseudowire** command does not work with devices running Cisco IOS Release 12.0(27)S1 or Release 12.0(27)S2. |
| 4 | • Version 8 (draft-ietf-mpls-ping-08)—Applicable before Cisco IOS Release 12.4(11)T. All echo packet's TLVs are formatted as specified in Version 8.<br>• RFC 4379 compliant—Applicable after Cisco IOS Release 12.4(11)T. All echo packet's TLVs are formatted as specified in RFC 4379.<br>This is the recommended version. |

1. If you do not specify the **revision** keyword, the software uses the latest version.

## Examples

The following example shows how to trace packets through an MPLS LDP LSP:

```
Router# trace mpls ipv4 10.131.191.252/32
```

Alternatively, you can use the interactive mode:

```
Protocol [ip]: mpls
Target IPv4, pseudowire or traffic-eng [ipv4]: <ipv4 |pseudowire |tunnel> ipv4
Target IPv4 address: 10.131.191.252
Target mask: /32
```

```
Repeat [1]:
Packet size [100]:
Timeout in seconds [2]:
Extended commands? [no]: yes
Destination start address:
Destination end address:
Source address:
EXP bits in mpls header [0]:
TimeToLive [255]:
Reply mode (2-ipv4 via udp, 3-ipv4 via udp with router alert) [2]:
Reply ip header DSCP bits [0]:

Tracing MPLS Label Switched Path to 10.131.191.252/32, timeout is 2 seconds

Codes:
    '!' - success, 'Q' - request not sent, '.' - timeout,
    'L' - labeled output interface, 'B' - unlabeled output interface,
    'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
    'M' - malformed request, 'm' - unsupported tlvs, 'N' - no rx label,
    'P' - no rx intf label prot, 'p' - premature termination of LSP,
    'R' - transit router, 'X' - unknown return code, 'x' - return code 0


Type escape sequence to abort.

  0 10.131.159.245 mtu 1500 []
! 1 10.131.191.252 100 ms
```

The following example shows how to trace packets through an MPLS TE tunnel:

```
Router# trace mpls traffic-eng Tunnel 0

Tracing MPLS TE Label Switched Path on Tunnel0, timeout is 2 seconds

Codes:
    '!' - success, 'Q' - request not sent, '.' - timeout,
    'L' - labeled output interface, 'B' - unlabeled output interface,
    'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
    'M' - malformed request, 'm' - unsupported tlvs, 'N' - no rx label,
    'P' - no rx intf label prot, 'p' - premature termination of LSP,
    'R' - transit router, 'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.

  0 10.131.159.230 mtu 1500 [Labels: 22 Exp: 0]
R 1 10.131.159.225 mtu 1500 [Labels: 22 Exp: 6] 72 ms
R 2 10.131.191.229 mtu 1504 [implicit-null] 72 ms
! 3 10.131.191.252 92 ms
```

Alternatively, you can use the interactive mode:

```
Router# traceroute
Protocol [ip]: mpls
Target IPv4 or tunnel [ipv4]: traffic-eng
Tunnel number [0]:
Repeat [1]:
Timeout in seconds [2]:
Extended commands? [no]:
Tracing MPLS TE Label Switched Path on Tunnel0, timeout is 2 seconds

Codes:
    '!' - success, 'Q' - request not sent, '.' - timeout,
    'L' - labeled output interface, 'B' - unlabeled output interface,
    'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
```

```
    'M' - malformed request, 'm' - unsupported tlvs, 'N' - no rx label,
    'P' - no rx intf label prot, 'p' - premature termination of LSP,
    'R' - transit router, 'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.

  0 10.131.159.230 mtu 1500 [Labels: 22 Exp: 0]
R 1 10.131.159.225 mtu 1500 [Labels: 22 Exp: 6] 72 ms
R 2 10.131.191.229 mtu 1504 [implicit-null] 72 ms
! 3 10.131.191.252 92 ms
```

Use the **show running-config** command to verify the configuration of Tunnel 0 (shown in bold). The tunnel destination has the same IP address as the one in the earlier trace IPv4 example, but the trace takes a different path, even though tunnel 0 is not configured to forward traffic by means of autoroute or static routing. The **trace mpls traffic-eng** command is powerful; it enables you to test the tunnels to verify that they work before you map traffic onto them.

```
Router# show running-config interface tunnel 0

Building configuration...

Current configuration : 210 bytes
!
interface Tunnel0
 ip unnumbered Loopback0
 no ip directed-broadcast
 tunnel destination 10.131.191.252       <---- Tunnel destination IP address.
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng path-option 5 explicit name as1pe-long-path
end

Router# show mpls traffic-eng tunnels tunnel 0 brief

Signalling Summary:
    LSP Tunnels Process:            running
    RSVP Process:                   running
    Forwarding:                     enabled
    Periodic reoptimization:        every 3600 seconds, next in 1369 seconds
    Periodic FRR Promotion:         Not Running
    Periodic auto-bw collection:    disabled
TUNNEL NAME                DESTINATION      UP IF     DOWN IF   STATE/PROT
PE_t0                      10.131.191.252   -         Et0/0     up/up

Router# show ip cef 10.131.191.252

10.131.191.252/32, version 37, epoch 0, cached adjacency 10.131.159.246
0 packets, 0 bytes
  tag information set, all rewrites owned
    local tag: 21
  via 10.131.159.246, Ethernet1/0, 0 dependencies
    next hop 10.131.159.246, Ethernet1/0
    valid cached adjacency
    tag rewrite with Et1/0, 10.131.159.246, tags imposed {}
```

The following example performs a trace operation on a multisegment pseudowire. The trace operation goes to segment 2 of the multisegment pseudowire.

```
Router# trace mpls pseudowire 10.10.10.9 220 segment 2

Tracing MS-PW segments within range [1-2] peer address 10.10.10.9 and timeout 2 seconds

Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
```

```
  'L' - labeled output interface, 'B' - unlabeled output interface,
  'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
  'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
  'P' - no rx intf label prot, 'p' - premature termination of LSP,
  'R' - transit router, 'I' - unknown upstream index,
  'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.
L 1 10.10.9.9 4 ms [Labels: 18 Exp: 0]
    local 10.10.10.22 remote 10.10.10.9 vc id 220

! 2 10.10.3.3 4 ms [Labels: 16 Exp: 0]
    local 10.10.10.9 remote 10.10.10.3 vc id 220
```

## Related Commands

| Command | Description |
|---|---|
| **ping mpls** | Checks MPLS LSP connectivity. |

# trace mpls multipath

To discover all Multiprotocol Label Switching (MPLS) label switched paths (LSPs) from an egress router to an ingress router, use the **trace mpls multipath** command in privileged EXEC mode.

**trace mpls multipath ipv4** *destination-address***/***destination-mask-length*
[**timeout** *seconds*]
[**interval** *milliseconds*]
[**destination** *address-start address-end*]
[**source** *source-address*]
[**exp** *exp-bits*]
[**ttl** *maximum-time-to-live*]
[**reply mode** {**ipv4** | **router-alert**}]
[**reply dscp** *dscp-value*]
[**retry-count** *retry-count-value*]
[**force-explicit-null**]
[**output interface** *tx-interface* [**nexthop** *ip-address*]]
[**hashkey ipv4 bitmap** *bitmap-size*]
[**flags fec**]
[**verbose**]

## Syntax Description

| | |
|---|---|
| **ipv4** | Specifies the destination type as a Label Distribution Protocol (LDP) IPv4 address. |
| *destination-address* | Address prefix of the target to be tested. |
| */destination-mask-length* | Number of bits in the network mask of the target address. The slash is required. |
| **timeout** *seconds* | (Optional) Specifies the timeout interval in seconds. The range is from 0 to 3600. The default is 2 seconds. |
| **interval** *milliseconds* | (Optional) Sets the time between successive MPLS echo requests in milliseconds. This allows you to pace the transmission of packets so that the receiving router does not drop packets. The default is 0 milliseconds. Valid values are from 0 to 3500000 milliseconds. |
| **destination** | (Optional) Specifies a network 127 address. |
| *address-start* | (Optional) The beginning network 127 address. |
| *address-end* | (Optional) The ending network 127 address. |
| **source** | (Optional) Specifies the source address or name. |
| *source-address* | (Optional) Source address or name. |
| **exp** *exp-bits* | (Optional) Specifies the MPLS experimental field value in the MPLS header for an MPLS echo reply. Valid values are from 0 to 7. Default is 0. |
| **ttl** *maximum-time-to-live* | (Optional) Specifies a maximum hop count. |
| **reply mode** {**ipv4** \| **router-alert**} | (Optional) Specifies the reply mode for the echo request packet.<br>The reply mode is one of the following:<br>• **ipv4** = Reply with an IPv4 User Datagram Protocol (UDP) packet (default).<br>• **router-alert** = Reply with an IPv4 UDP packet with router alert. |

| | |
|---|---|
| **reply dscp** *dscp-value* | (Optional) Controls the differentiated services codepoint (DSCP) value of an echo reply. Allows the support of a class of service (CoS) in an echo reply. |
| **retry-count** re*try-count-value* | (Optional) Sets the number of timeout retry attempts during a multipath LSP trace. A retry is attempted if an outstanding echo request times out waiting for the corresponding echo reply.<br>A re*try-count-value* of 0 means infinite retries. Valid values are from 0 to 10. |
| **force-explicit-null** | (Optional) Forces an explicit null label to be added to the MPLS label stack even though the label was unsolicited. |
| **output interface** *tx-interface* | (Optional) Specifies the output interface for MPLS echo requests. |
| **nexthop** *ip-address* | (Optional) Causes packets to go through the specified next hop address. |
| **hashkey ipv4 bitmap** *bitmap-size* | (Optional) Allows you to control the hash key and multipath settings.<br>• **ipv4**—Indicates an IPv4 address, which is the only hashkey type valid for multipath (type 8).<br>• **bitmap** *bitmap-size*—Size of the bitmap IPv4 addresses. |
| **flags fec** | (Optional) Requests that target Forwarding Equivalence Class (FEC) stack validation of a transit router be done at the egress router.<br>**Note** Be sure to use the **flags fec** keywords in conjunction with the **ttl** keyword. |
| **verbose** | (Optional) Displays the MPLS echo reply sender address of the packet and displays return codes. |

## Command Default

**timeout** = 2 seconds
**interval** = 0 milliseconds
**reply mode** = IPv4 via UDP (2)
Maximum time-to-live = 30 hops
Experimental bits in MPLS header = 0

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| 12.2(31)SB2 | This command was introduced. |
| 12.2(33)SRB | This command was integrated into Cisco IOS Release 12.2(33)SRB. |
| 12.4(20)T | This command was integrated into Cisco IOS Release 12.4(20)T. |

## Usage Guidelines

Use the **trace mpls multipath** command to discover all possible paths between an egress and ingress router in multivendor networks that use IPv4 load balancing at the transit routers.

Use the **destination** *address-start address-end* keyword and arguments to specify a valid 127/8 address. You have the option to specify a single *x.y.z-address* or a range of numbers from 0.0.0 to *x.y.z*, where *x*, *y*, and *z* are numbers from 0 to 255 and correspond to the 127.*x.y.z* destination address. The MPLS echo request destination address in the UDP packet is not used to forward the MPLS packet to the destination router. The label stack that is used to forward the echo request routes the MPLS packet to the destination router. The 127/8 address guarantees that the packets are routed to the localhost (the default loopback address of the router processing the address) if the UDP packet destination address is used for forwarding. In addition, the destination address is used to adjust load balancing when the destination address of the IP payload is used for load balancing.

## Examples

The following example shows how to discover all IPv4 LSPs to a router whose IP address is 10.1.1.150:

```
Router# trace mpls multipath ipv4 10.1.1.150/32

Starting LSP Multipath Traceroute for 10.1.1.150/32

Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
  'L' - labeled output interface, 'B' - unlabeled output interface,
  'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
  'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
  'P' - no rx intf label prot, 'p' - premature termination of LSP,
  'R' - transit router, 'I' - unknown upstream index,
  'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.
LLLL!
Path 0 found,
 output interface Et0/0 source 10.1.111.101 destination 127.0.0.0 LLL!
Path 1 found,
 output interface Et0/0 source 10.1.111.101 destination 127.0.0.1 L!
Path 2 found,
 output interface Et0/0 source 10.1.111.101 destination 127.0.0.5 LL!
Path 3 found,
 output interface Et0/0 source 10.1.111.101 destination 127.0.0.7
Paths (found/broken/unexplored) (4/0/0)
  Echo Request (sent/fail) (14/0)
  Echo Reply (received/timeout) (14/0)
  Total Time Elapsed 472 ms
```

The following example shows how to set the number of timeout retry attempts to 4 during a multipath LSP trace:

```
Router# trace mpls multipath ipv4 10.1.1.150/32 retry-count 4

Starting LSP Multipath Traceroute for 10.1.1.150/32

Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
  'L' - labeled output interface, 'B' - unlabeled output interface,
  'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
  'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
  'P' - no rx intf label prot, 'p' - premature termination of LSP,
  'R' - transit router, 'I' - unknown upstream index,
  'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.
LLLL!
Path 0 found,
 output interface Et0/0 source 10.1.111.101 destination 127.0.0.0 LLL!
Path 1 found,
 output interface Et0/0 source 10.1.111.101 destination 127.0.0.1 L!
Path 2 found,
 output interface Et0/0 source 10.1.111.101 destination 127.0.0.5 LL!
Path 3 found,
 output interface Et0/0 source 10.1.111.101 destination 127.0.0.7
```

```
Paths (found/broken/unexplored) (4/0/0)
 Echo Request (sent/fail) (14/0)
 Echo Reply (received/timeout) (14/0)
 Total Time Elapsed 460 ms
```

The following example shows that outgoing MPLS Operation, Administration, and Management (OAM) echo request packets will go through the interface e0/0 and will be restricted to the path with the next hop address of 10.0.0.3:

```
Router# trace multipath ipv4 10.4.4.4/32 output interface e0/0 nexthop 10.0.0.3

Starting LSP Multipath Traceroute for 10.4.4.4/32

Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
  'L' - labeled output interface, 'B' - unlabeled output interface,
  'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
  'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
  'P' - no rx intf label prot, 'p' - premature termination of LSP,
  'R' - transit router, 'I' - unknown upstream index,
  'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.
L!
Path 0 found,
 output interface Et0/0 nexthop 10.0.0.3
 source 10.0.0.1 destination 127.0.0.0

Paths (found/broken/unexplored) (1/0/0)
 Echo Request (sent/fail) (2/0)
 Echo Reply (received/timeout) (2/0)
 Total Time Elapsed 728 ms
```

## Related Commands

| Command | Description |
|---|---|
| **echo** | Customizes the default behavior of echo packets. |
| **mpls oam** | Enters MPLS OAM configuration mode for customizing the default behavior of echo packet. |
| **ping mpls** | Checks MPLS LSP connectivity. |
| **trace mpls** | Discovers MPLS LSP routes that packets will actually take when traveling to their destinations. |

# trace mpls tp

To display the Multiprotocol Label Switching (MPLS) transport protocol (TP) label switched path (LSP) routes that packets take to their destinations, use the **trace mpls tp** command in privileged EXEC mode.

**trace mpls tp tunnel-tp** *num* **lsp** {**working** | **protect** | **active**}
[**destination** *ip-addr*]
[**exp** *num*]
[**flags fec** ]
[**reply dscp** *num* | **mode control channel** ]
[**source** *ip-addr*]
[**timeout** *num*]
[**ttl** *num*]
[**verbose**]

## Syntax Description

| | |
|---|---|
| **tunnel-tp** *num* | The MPLS-TP tunnel number. |
| **lsp** {**working** \| **protect** \| **active**} | Specifies the type of MPLS-TP label switched path (LSP) on which to send echo request packets. |
| **destination** *ip-addr* | (Optional) Specifies a network 127 address. |
| **exp** *num* | (Optional) Specifies the MPLS experimental field value in the MPLS header for an MPLS echo reply. Valid values are from 0 to 7. The Default is 0. |
| **flags fec** | (Optional) Allows Forward Equivalence Class (FEC) checking on the transit router. A downstream map type-length-value (TLV) containing the correct received labels must be present in the echo request for target FEC stack checking to be performed.<br>Target FEC stack validation is always done at the egress router. Be sure to use this keyword in conjunction with the **ttl** keyword. |
| **reply dscp** *num* \| **mode control channel** | (Optional) Provides the capability to request a specific quality of service (QoS) in an echo reply by providing a differentiated services code point (DSCP) value.<br>The echo reply is returned with the IP header type of service (ToS) byte set to the value specified in the **reply dscp** command. |
| **source** *ip-addr* | (Optional) Specifies the source address or name. The default address is loopback0. This address is used as the destination address in the MPLS echo response. |
| **timeout** *num* | (Optional) Specifies the timeout interval in seconds for an MPLS request packet. The range is from 0 to 3600. The default is 2. |
| **ttl** *num* | (Optional) Specifies a time-to-live (TTL) value. The default is 225. |
| **verbose** | (Optional) Enables verbose output mode. |

## Command Default

Connectivity is not checked.

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| 15.1(1)SA | This command was introduced. |
| 15.1(3)S | This command was integrated. |

## Usage Guidelines

Use the **trace mpls tp** command to validate, test, or troubleshoot MPLS TP LSPs.

**Note** The **trace mpls tp** command does not support interactive mode.

You can use ping and trace in an MPLS-TP network without IP addressing. However, no IP addresses are displayed in the output.

The following rules determine the source IP address:

1. Use the IP address of the TP interface/
2. Use the global router ID.
3. Use router ID : A.B.C.D local node ID in IPv4 address format. This is not an IP address however it is better to use a value rather than leave it as 0.0.0.0 and risk the packet being deemed invalid and dropped.

## Examples

The following example checks connectivity of an MPLS-TP LSP:

```
Router# trace mpls tp tunnel-tp 1 lsp working verbose

Tracing MPLS TP Label Switched Path on Tunnel-tp1, timeout is 2 seconds

Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
  'L' - labeled output interface, 'B' - unlabeled output interface,
  'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
  'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
  'P' - no rx intf label prot, 'p' - premature termination of LSP,
  'R' - transit router, 'I' - unknown upstream index,
  'l' - Label switched with FEC change, 'd' - see DDMAP for return code,
  'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.
  0 1.1.1.5 127.0.0.1 MRU 1500 [Labels: 444 Exp: 0]
I 1 0.0.0.0 127.0.0.1 MRU 1500 [Labels: 300/13 Exp: 0/0] 1 ms, ret code 6
! 2 0.0.0.0 1 ms, ret code 3
```

| Command | Description |
|---|---|
| **ping mpls tp** | Checks MPLS-TP LSP connectivity. |

# traffic-engineering filter

To specify a filter with the given number and properties, use the **traffic-engineering filter** command in router configuration mode. To disable this function, use the **no** form of this command.

**traffic-engineering filter** *filter-number* **egress** *ip-address mask*

**no traffic-engineering filter**

## Syntax Description

| | |
|---|---|
| *filter-number* | A decimal value representing the number of the filter. |
| **egress** *ip-address mask* | IP address and mask for the egress port. |

## Defaults

Disabled

## Command Modes

Router configuration

## Command History

| Release | Modification |
|---|---|
| 11.1 CT | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

You must specify that the egress is the indicated address or mask, where egress is either the destination or the Border Gateway Protocol (BGP) next hop.

## Examples

The following example shows how to configure a traffic engineering filter and a traffic engineering route for that filter over a label switched path (LSP)-encapsulated tunnel for the traffic engineering routing process:

```
Router(config)# router traffic-engineering
Router(config-router)# traffic-engineering filter 5 egress 10.0.0.1 255.255.255.255
Router(config-router)# traffic-engineering route 5 tunnel 5
```

## Related Commands

| Command | Description |
|---|---|
| **show ip traffic-engineering routes** | Displays information about the requested filters configured for traffic engineering. |
| **traffic-engineering route** | Configures a route for a specified filter, through a specified tunnel. |

# traffic-engineering route

To configure a route for a specified filter through a specified tunnel, use the **traffic-engineering route** command in router configuration mode. To disable this function, use the **no** form of this command.

**traffic-engineering route** *filter-number interface* [**preference** *number*] [**loop-prevention** {**on** | **off**}]

**no traffic-engineering route** *filter-number interface* [**preference** *number*] [**loop-prevention** {**on** | **off**}]

## Syntax Description

| | |
|---|---|
| *filter-number* | The number of the traffic engineering filter to be forwarded through the use of this traffic engineering route, if the route is installed. |
| *interface* | Label switched path (LSP)-encapsulated tunnel on which the traffic-passing filter should be sent, if this traffic engineering route is installed. |
| **preference** *number* | (Optional) This is a number from 1 to 255, with a lower value being more desirable. The default is 1. |
| **loop-prevention** | (Optional) A setting of **on** or **off**. The default is **on**. |

## Defaults

**preference**: 1
**loop-prevention**: on

## Command Modes

Router configuration

## Command History

| Release | Modification |
|---|---|
| 11.1 CT | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

The traffic engineering process is used to decide if a configured traffic engineering route should be installed in the forwarding table.

The first step is to determine if the route is up. If the route is enabled, the LSP tunnel interface is up, the loop prevention check is either disabled or passed, and the traffic engineering route is up.

If multiple routes for the same filter are up, a route is selected based on administrative preference.

If loop prevention is enabled, metrics are solicited from the tunnel tail, and the loop prevention algorithm is run on the result. For a discussion of the loop prevention algorithm, see the **show ip traffic-engineering metrics** command.

## Examples

The following example shows how to configure a traffic engineering filter and a traffic engineering route for that filter through an LSP-encapsulated tunnel for the traffic engineering routing process:

```
Router(config)# router traffic-engineering
Router(config-router)# traffic-engineering filter 5 egress 10.0.0.1 255.255.255.255
Router(config-router)# traffic-engineering route 5 tunnel 5
```

## Related Commands

| Command | Description |
|---|---|
| **show ip traffic-engineering configuration** | Displays information about configured traffic engineering filters and routes. |
| **show ip traffic-engineering routes** | Displays information about the requested filters configured for traffic engineering. |

# transport vpls mesh

To create a full mesh of pseudowires under a virtual private LAN switching (VPLS) domain, use the **transport vpls mesh** command in interface configuration mode. To remove the mesh of pseudowires, use the **no** form of this command.

**transport vpls mesh**

**no transport vpls mesh**

## Syntax Description

This command has no arguments or keywords.

## Command Default

The transport type is not specified.

## Command Modes

Interface configuration (config-if)#)

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SXI4 | This command was introduced. |

## Usage Guidelines

This command creates a full mesh of pseudowires under a VPLS domain.

## Examples

The following example creates a virtual Ethernet interface and then specifies a full mesh of pseudowires:

```
Router(config)# interface virtual-ethernet 1
Router(config-if)# transport vpls mesh
```

## Related Commands

| Command | Description |
|---|---|
| **interface virtual-ethernet** | Create a virtual Ethernet interfaces |

# tunnel destination access-list

To specify the access list that the template interface uses for obtaining the mesh tunnel interface destination address, use the **tunnel destination access-list** command in interface configuration mode. To remove the access list from this template interface, use the **no** form of this command.

**tunnel destination access-list** *num*

**no tunnel destination access-list** *num*

## Syntax Description

| | |
|---|---|
| *num* | Number of the access list. |

## Command Default

No default behavior or values to specify access lists.

## Command Modes

Interface configuration (config-if)#

## Command History

| Release | Modification |
|---|---|
| 12.0(27)S | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2(33)SXH | This command was integrated into Cisco IOS Release 12.2(33)SXH. |
| 12.4(20)T | This command was integrated into Cisco IOS Release 12.4(20)T. |

## Usage Guidelines

This command can be used only on template interfaces.

If you specify an access list that does not exist, no tunnels are set up. You need an access list to set up the destination addresses for the mesh tunnel interfaces.

If you enter the **shutdown** command on the autotemplate interface, the command is executed on all the cloned tunnel interfaces. To delete all the cloned tunnel interfaces, enter the **no tunnel destination** command on the autotemplate. To delete tunnel interfaces for a particular autotemplate, go to the particular interface and enter the **no tunnel destination** command.

## Examples

The following example shows how to configure the template interface to use access-list 1 to obtain the tunnel destination address:

```
Router(config)# interface auto-template 1
Router(config-if)# tunnel destination access-list 1
```

## Related Commands

| Command | Description |
|---|---|
| **interface auto-template** | Creates the template interface. |
| **mpls traffic-eng auto-tunnel mesh tunnel-num** | Configures a range of mesh tunnel interface numbers. |

# tunnel destination list mpls traffic-eng

To specify a list of Multiprotocol Label Switching (MPLS) traffic engineering (TE) point-to-multipoint (P2MP) destinations, use the **tunnel destination list mpls traffic-eng** command in interface configuration mode. To remove the destination list, use the **no** form of this command.

**tunnel destination list mpls traffic-eng** {**id** *destination-list-number* | **name** *destination-list-name*}

**no tunnel destination list mpls traffic-eng** {**id** *dest-list-number* | **name** *dest-list-name*}

## Syntax Description

| | |
|---|---|
| **id** *destination-list-identifier* | Specifies the number of a destination list. Valid range of numbers is 1–65535. |
| **name** *destination-list-name* | Specifies the name of a destination list. |

## Command Default

No destination list is specified.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SRE | This command was introduced. |

## Usage Guidelines

Use the **tunnel destination list mpls traffic-eng** command to specify a list point-to-multipoint tunnels.

## Examples

The following example configures point-to-multipoint traffic engineering on tunnel interface 1:

```
Router# interface tunnel1
Router(config-if)# tunnel mode mpls traffic-eng point-to-multipoint
Router(config-if)# tunnel destination list mpls traffic-eng name P2MP-DYN-DST-LIST
```

## Related Commands

| Command | Description |
|---|---|
| **show mpls traffic-eng tunnels** | Displays MPLS TE tunnels. |
| **tunnel destination list mpls traffic-eng** | Specifies the list of MPLS TE P2MP destinations. |

# tunnel destination mesh-group

To specify a mesh group that an autotemplate interface uses to signal tunnels for all mesh group members, use the **tunnel destination mesh group** command in interface configuration mode. To remove a mesh group from the template, use the **no** form of this command.

**tunnel destination mesh-group** *mesh-group-id*

**no tunnel destination mesh-group** *mesh-group-id*

## Syntax Description

| | |
|---|---|
| *mesh-group-id* | Number that identifies a specific mesh group. |

## Command Default

Mesh-groups are not advertised.

## Command Modes

Interface configuration (config-if)#

## Command History

| Release | Modification |
|---|---|
| 12.0(29)S | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2(33)SXH | This command was integrated into Cisco IOS Release 12.2(33)SXH. |
| 12.4(20)T | This command was integrated into Cisco IOS Release 12.4(20)T. |

## Usage Guidelines

Use this command to associate a specific mesh group with an autotemplate. When a mesh group is associated with an autotemplate, the template interface signals tunnels for all mesh group members.

## Examples

The following example shows how to configure an autotemplate to signal tunnels for mesh group 10:

```
Router(config)# interface auto-template 1
Router(config-if)# tunnel destination mesh-group 10
```

## Related Commands

| Command | Description |
|---|---|
| **mpls traffic-eng mesh-group** | Configures an IGP to allow MPLS TE LSRs that belong to the same mesh group to signal tunnels to the local router. |

# tunnel flow egress-records

To create a NetFlow record for packets that are encapsulated by a generic routing encapsulation (GRE) tunnel when both NetFlow and Cisco Express Forwarding are enabled, use the **tunnel flow egress-records** command in interface configuration mode. To disable NetFlow record creation, use the **no** form of this command.

**tunnel flow egress-records**

**no tunnel flow egress-records**

## Syntax Description

This command has no arguments or keywords.

## Defaults

A NetFlow record for encapsulated packets is not created.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.2(2)T | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

When this command is enabled on a GRE tunnel with both Cisco Express Forwarding and NetFlow enabled, a NetFlow record is created for packets that are encapsulated by the tunnel.

## Examples

The following example shows how to enable NetFlow record creation:

```
Router(config-if)# tunnel flow egress-records
```

## Related Commands

| Command | Description |
|---|---|
| **show ip cache flow** | Displays NetFlow switching statistics. |

# tunnel mode mpls traffic-eng

To set the mode of a tunnel to Multiprotocol Label Switching (MPLS) for traffic engineering, use the **tunnel mode mpls traffic-eng** command in interface configuration mode. To disable this feature, use the **no** form of this command.

**tunnel mode mpls traffic-eng**

**no tunnel mode mpls traffic-eng**

## Syntax Description

This command has no arguments or keywords.

## Defaults

Disabled.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.0(5)S | This command was introduced. |
| 12.2(28)SB | This command was integrated into Cisco IOS Release 12.2(28)SB. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

This command specifies that the tunnel interface is for an MPLS traffic engineering tunnel and enables the various tunnel MPLS configuration options.

## Examples

The following example shows how to set the mode of the tunnel to MPLS traffic engineering:

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

## Related Commands

| Command | Description |
|---|---|
| **tunnel mpls traffic-eng affinity** | Configures an affinity for an MPLS traffic engineering tunnel. |
| **tunnel mpls traffic-eng autoroute announce** | Instructs the IGP to use the tunnel in its enhanced SPF algorithm calculation (if the tunnel is up). |
| **tunnel mpls traffic-eng bandwidth** | Configures the bandwidth required for an MPLS traffic engineering tunnel. |

| Command | Description |
|---|---|
| **tunnel mpls traffic-eng path-option** | Configures a path option. |
| **tunnel mpls traffic-eng priority** | Configures setup and reservation priority for an MPLS traffic engineering tunnel. |

# tunnel mode mpls traffic-eng point-to-multipoint

To enable the configuration of a Multiprotocol Label Switching (MPLS) traffic engineering (TE) point-to-multipoint (P2MP) tunnel, use the **tunnel mode mpls traffic-eng point-to-multipoint** command in interface configuration mode. To remove the tunnel, use the **no** form of this command.

**tunnel mode mpls traffic-eng point-to-multipoint**

**no tunnel mode**

## Syntax Description

This command has no arguments or keywords.

## Command Default

No point-to-multipoint tunnel mode is enabled.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SRE | This command was introduced. |

## Usage Guidelines

Use the command to differentiate point-to-multipoint tunnels from point-to-point tunnels.

## Examples

The following example configures point-to-multipoint traffic engineering on tunnel interface 1:

```
Router# interface Tunnel1
Router(config-if)# tunnel mode mpls traffic-eng point-to-multipoint
Router(config-if)# tunnel destination list mpls traffic-eng name P2MP-DYN-DST-LIST
```

## Related Commands

| Command | Description |
|---|---|
| **show mpls traffic-eng tunnels** | Displays MPLS TE tunnels. |
| **tunnel destination list mpls traffic-eng** | Specifies the list of MPLS TE P2MP destinations. |

# tunnel mpls traffic-eng affinity

To configure an affinity (the properties the tunnel requires in its links) for a Multiprotocol Label Switching (MPLS) traffic engineering tunnel, use the **tunnel mpls traffic-eng affinity** command in interface configuration mode. To disable the MPLS traffic engineering tunnel affinity, use the **no** form of this command.

**tunnel mpls traffic-eng affinity** *properties* [**mask** *mask value*]

**no tunnel mpls traffic-eng affinity** *properties* [**mask** *mask value*]

## Syntax Description

| | |
|---|---|
| *properties* | Attribute values required for links carrying this tunnel. A 32-bit decimal number. Valid values are from 0x0 to 0xFFFFFFFF, representing 32 attributes (bits), where the value of an attribute is 0 or 1. |
| **mask** *mask value* | (Optional) Link attribute to be checked. A 32-bit decimal number. Valid values are from 0x0 to 0xFFFFFFFF, representing 32 attributes (bits), where the value of an attribute is 0 or 1. |

## Defaults

*properties*: 0X00000000
*mask value*: 0X0000FFFF

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.0(5)S | This command was introduced. |
| 12.2(28)SB | This command was integrated into Cisco IOS Release 12.2(28)SB. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

The affinity determines the attributes of the links that this tunnel will use (that is, the attributes for which the tunnel has an affinity). The attribute mask determines which link attribute the router should check. If a bit in the mask is 0, an attribute value of a link or that bit is irrelevant. If a bit in the mask is 1, the attribute value of a link and the required affinity of the tunnel for that bit must match.

A tunnel can use a link if the tunnel affinity equals the link attributes and the tunnel affinity mask.

Any properties set to 1 in the affinity should also be 1 in the mask. In other words, affinity and mask should be set as follows:

```
tunnel_affinity = (tunnel_affinity and tunnel_affinity_mask)
```

## Examples

The following example shows how to set the affinity of the tunnel to 0x0101 mask 0x303:

```
Router(config-if)# tunnel mpls traffic-eng affinity 0x0101 mask 0x303
```

## Related Commands

| Command | Description |
|---|---|
| **mpls traffic-eng attribute-flags** | Sets the attributes for the interface. |
| **tunnel mode mpls traffic-eng** | Sets the mode of a tunnel to MPLS for traffic engineering. |

# tunnel mpls traffic-eng autoroute destination

To automatically route traffic through a traffic engineering (TE) tunnel, use the **tunnel mpls traffic-eng autoroute destination** command in interface configuration mode. To disable this feature, use the **no** form of this command.

**tunnel mpls traffic-eng autoroute destination**

**no tunnel mpls traffic-eng autoroute destination**

## Syntax Description

This command has no arguments or keywords.

## Command Default

If you do not enter this command, manually-configured static routes are required.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SRE | This command was introduced. |

## Usage Guidelines

The **tunnel mpls traffic-eng autoroute destination** command prevents you from having to manually configure static routes. Use the **tunnel mpls traffic-eng autoroute destination** command because interarea TE tunnels cross areas.

For interarea tunnels, the **tunnel mpls traffic-eng autoroute announce** command and the **tunnel mpls traffic-eng forwarding-adjacency** command are not operational.

## Examples

The following example specifies that tunnel 103 has autoroute destination enabled:

```
Router(config)# interface Tunnel103
Router(config-if)# ip unnumbered Loopback0
Router(config-if)# tunnel destination 10.1.0.3
Router(config-if)# tunnel mode mpls traffic-eng
Router(config-if)# tunnel mpls traffic-eng autoroute destination
Router(config-if)# tunnel mpls traffic-eng path-option 1 explicit name 111-103
```

## Related Commands

| Command | Description |
|---|---|
| **tunnel mpls traffic-eng autoroute announce** | Specifies that the IGP should use the tunnel (if the tunnel is up) in its enhanced SPF calculation. |
| **tunnel mpls traffic-eng forwarding-adjacency** | Advertises a TE tunnel as a link in an IGP network. |

# tunnel mpls traffic-eng auto-bw

To configure a tunnel for automatic bandwidth adjustment and to control the manner in which the bandwidth for a tunnel is adjusted, use the **tunnel mpls traffic-eng auto-bw** command in interface configuration mode. To disable automatic bandwidth adjustment for a tunnel, use the **no** form of this command.

**tunnel mpls traffic-eng auto-bw** [**collect-bw**] [**frequency** *seconds*] [**max-bw** *number*] [**min-bw** *number*]

**no tunnel mpls traffic-eng auto-bw**

## Syntax Description

| | |
|---|---|
| **collect-bw** | (Optional) Collects output rate information for the tunnel, but does not adjust the tunnel's bandwidth. |
| **frequency** *seconds* | (Optional) The interval between bandwidth adjustments. The specified interval can be from 300 to 604800 seconds. Do not specify a value lower than the output rate sampling interval specified in the **mpls traffic-eng auto-bw** command. |
| **max-bw** *number* | (Optional) Maximum automatic bandwidth, in kbps, for this tunnel. The value is from 0 to 4294967295. |
| **min-bw** *number* | (Optional) Minimum automatic bandwidth, in kbps, for this tunnel. The value is from 0 to 4294967295. For information about the default, see "Usage Guidelines." |

## Command Default

You cannot control the manner in which the bandwidth for a tunnel is adjusted.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.2(4)T | This command was introduced. |
| 12.2(11)S | This command was integrated into Cisco IOS Release 12.2(11)S. |
| 12.2(14)S | This command was integrated into Cisco IOS Release 12.2(14)S. |
| 12.2(28)SB | This command was integrated into Cisco IOS Release 12.2(28)SB. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

If you enter the command with no optional keywords or arguments, automatic bandwidth adjustment for the tunnel is enabled, with adjustments made every 24 hours and with no constraints on the bandwidth adjustment made.

To sample the bandwidth used by a tunnel without automatically adjusting it, specify the **collect-bw** keyword in the **tunnel mpls traffic-eng auto-bw** command.

If you do not specify the **collect-bw** keyword, the tunnel's bandwidth is adjusted to the largest average output rate sampled for the tunnel since the last bandwidth adjustment for the tunnel was made. If you do not specify the **collect-bw** keyword but you do enter some but not all of the other keywords, the defaults for the options not entered are: **frequency**, every 24hours; **min-bw**, unconstrained (0); and **max-bw**, unconstrained.

To constrain the bandwidth adjustment that can be made to a tunnel, use the **max-bw** or **min-bw** keyword and specify the permitted maximum allowable bandwidth or minimum allowable bandwidth, respectively.

The following rules apply to adjusting bandwidth on a tunnel:

- If the current bandwidth is less than 50 kbps, you can change the bandwidth only if the changed bandwidth is 10 kbps or more.
- If the current bandwidth is more than 50 kbps, you can change the bandwidth regardless of what percent it is of the current bandwidth.
- If the minimum or maximum bandwidth values are configured for a tunnel, the bandwidth stays between those values.
- If you configure a tunnel's bandwidth (in the **tunnel mpls traffic-eng bandwidth** command) and the minimum amount of automatic bandwidth (in the **tunnel mpls traffic-eng auto-bw** command), the minimum amount of automatic bandwidth adjustment is the lower of those two configured values. The default value of the **tunnel mpls traffic-eng bandwidth** command is 0.

The **no** form of the **tunnel mpls traffic-eng auto-bw** command disables bandwidth adjustment for the tunnel and restores the configured bandwidth for the tunnel bandwidth where "configured bandwidth" is determined as follows:

- If the tunnel bandwidth was explicitly configured via the **tunnel mpls traffic-eng bandwidth** command after the running configuration was written (if at all) to the startup configuration, the "configured bandwidth" is the bandwidth specified by that command.
- Otherwise, the "configured bandwidth" is the bandwidth specified for the tunnel in the startup configuration.

**Note** When you save the router configuration, the current bandwidth (not the originally configured bandwidth) is saved for tunnels with automatic bandwidth enabled.

**Note** Each **tunnel mpls traffic-eng auto-bw** command supersedes the previous one. Therefore, if you want to specify multiple arguments for a tunnel, you must specify them all in a single **tunnel mpls traffic-eng auto-bw** command.

**Note** Keywords for the **tunnel mpls traffic-eng auto-bw** command are order-dependent; you must enter them in the order in which they are listed in the command format.

## Examples

The following example shows how to enable automatic bandwidth adjustment for tunnel102 and specify that the adjustments are to occur every hour:

```
Router(config)# interface tunnel102
Router(config-if)# tunnel mpls traffic-eng auto-bw frequency 3600
```

## Related Commands

| Command | Description |
|---|---|
| **mpls traffic-eng auto-bw timers** | Enables automatic bandwidth adjustment on a platform for tunnels configured for bandwidth adjustment. |
| **tunnel mode mpls traffic-eng** | Sets the mode of a tunnel to MPLS for traffic engineering. |
| **tunnel mpls traffic-eng bandwidth** | Configures bandwidth required for an MPLS traffic engineering tunnel, |

# tunnel mpls traffic-eng autoroute announce

To specify that the Interior Gateway Protocol (IGP) should use the tunnel (if the tunnel is up) in its enhanced shortest path first (SPF) calculation, use the **tunnel mpls traffic-eng autoroute announce** command in interface configuration mode. To disable this feature, use the **no** form of this command.

**tunnel mpls traffic-eng autoroute announce**

**no tunnel mpls traffic-eng autoroute announce**

## Syntax Description

This command has no arguments or keywords.

## Command Default

The IGP does not use the tunnel in its enhanced SPF calculation.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 12.0(5)S | This command was introduced. |
| 12.2(28)SB | This command was integrated into Cisco IOS Release 12.2(28)SB. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |
| Cisco IOS XE Release 3.3S | This command was integrated into Cisco IOS XE Release 3.3S. |
| 15.2(1)T | This command was integrated into Cisco IOS Release 15.2(1)T. |

## Usage Guidelines

The only way to forward traffic onto a tunnel is by enabling this command or by explicitly configuring forwarding (for example, with an interface static route).

## Examples

The following example shows how to specify that the IGP should use the tunnel in its enhanced SPF calculation if the tunnel is up:

```
Router(config-if)# tunnel mpls traffic-eng autoroute announce
```

## Related Commands

| Command | Description |
|---|---|
| **ip route** | Establishes static routes. |
| **tunnel mode mpls traffic-eng** | Sets the mode of a tunnel to MPLS for traffic engineering. |

# tunnel mpls traffic-eng autoroute metric

To specify the Multiprotocol Label Switching (MPLS) traffic engineering tunnel metric that the Interior Gateway Protocol (IGP) enhanced shortest path first (SPF) calculation uses, use the **tunnel mpls traffic-eng autoroute metric** command in interface configuration mode. To disable the specified MPLS traffic engineering tunnel metric, use the **no** form of this command.

**tunnel mpls traffic-eng autoroute metric** {**absolute** | **relative**} *value*

**no tunnel mpls traffic-eng autoroute metric**

## Syntax Description

| | |
|---|---|
| **absolute** | Absolute metric mode; you can enter a positive metric value. |
| **relative** | Relative metric mode; you can enter a positive, negative, or zero value. |
| *value* | The metric that the IGP enhanced SPF calculation uses. The **relative** value can be from –10 to 10.<br>**Note** Even though the value for a relative metric can be from –10 to 10, configuring a tunnel metric with a negative value is considered a misconfiguration. If from the routing table the metric to the tunnel tail appears to be 4, then the cost to the tunnel tail router is actually 3 because 1 is added to the cost for getting to the loopback address. In this instance, the lowest value that you can configure for the relative metric is –3. |

## Defaults

The default is metric relative 0.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.0(5)S | This command was introduced. |
| 12.2(28)SB | This command was integrated into Cisco IOS Release 12.2(28)SB. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Examples

The following example shows how to specify the use of MPLS traffic engineering tunnel metric negative 1 for the IGP enhanced SPF calculation:

```
Router(config-if)# tunnel mpls traffic-eng autoroute metric relative -1
```

## Related Commands

| Command | Description |
| --- | --- |
| **show mpls traffic-eng autoroute** | Displays the tunnels announced to IGP, including interface, destination, and bandwidth. |
| **tunnel mpls traffic-eng autoroute announce** | Instructs the IGP to use the tunnel (if it is up) in its enhanced SPF calculation. |

# tunnel mpls traffic-eng backup-bw

To specify what types of label-switched paths (LSPs) can use a backup tunnel or whether the backup tunnel should provide bandwidth protection, and if so, how much, use the **tunnel mpls traffic-eng backup-bw** command in interface configuration mode.

**tunnel mpls traffic-eng backup-bw** {*kbps* | [**sub-pool** {*kbps* | **Unlimited**}] [**global-pool** {*kbps* | **Unlimited**}]} {*kbps* | [**class-type** {*kbps* | **Unlimited**}]

## Syntax Description

| | |
|---|---|
| *kbps* | Amount of bandwidth in kilobits per second (kbps), that this backup tunnel can protect. The router limits the number of LSPs that can use this backup tunnel so that the sum of the bandwidth of the LSPs does not exceed the specified amount of bandwidth. If there are multiple backup tunnels, the router will use the best-fit algorithm. |
| **sub-pool** | Only LSPs using bandwidth from the subpool can use the backup tunnel. |
| **global-pool** | Only LSPs using bandwidth from the global pool can use the backup tunnel. |
| **class-type** | Enter the class type. |
| **Unlimited** | Backup tunnel does not provide bandwidth protection. Any number of LSPs can use the backup tunnel, regardless of their bandwidth. |

## Command Default

If neither the **sub-pool** nor **global-pool** keyword is entered, any LSP (those using bandwidth from the subpool or global pool) can use this backup tunnel.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.0(22)S | This command was introduced. |
| 12.2(18)SXD1 | This command was integrated into Cisco IOS Release 12.2(18)SXD1. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.4(20)T | This command was integrated into Cisco IOS Release 12.4(20)T. |

## Usage Guidelines

If both the **sub-pool** and **global-pool** keywords are specified, **sub-pool** keyword must be specified first on the command line. For example, **tunnel mpls traffic-eng backup-bw sub-pool 100 global-pool Unlimited** is legal, but it is not legal to specify **tunnel mpls traffic-eng backup-bw global-pool Unlimited sub-pool 100**.

To limit the number of both subpool and global pool LSPs, enter the **tunnel mpls traffic-eng backup-bw sub-pool** *kbps* **global-pool** *kbps* command.

The **Unlimited** keyword cannot be used for both the subpool and global pool.

## Examples

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

```
Router(config)# interface Tunnel1
Router(config-if)# tunnel mpls traffic-eng backup-bw global-pool Unlimited
Router(config-if)# end

Router(config)# interface Tunnel2
Router(config-if)# tunnel mpls traffic-eng backup-bw sub-pool 1000
Router(config-if)# end
```

## Related Commands

| Command | Description |
| --- | --- |
| **mpls traffic-eng backup path** | Assigns one or more backup tunnels to a protected interface. |

# tunnel mpls traffic-eng bandwidth

To configure the bandwidth required for a Multiprotocol Label Switching (MPLS) traffic engineering (TE) tunnel, use the **tunnel mpls traffic-eng bandwidth** command in interface configuration mode. To disable this bandwidth configuration, use the **no** form of this command.

**tunnel mpls traffic-eng bandwidth** {*kbps* [**class-type** *value*] | **sub-pool** *kbps*}

**no tunnel mpls traffic-eng bandwidth**

## Syntax Description

| | |
|---|---|
| **sub-pool** | (Optional) Indicates a subpool tunnel. |
| **class-type** | (Optional) IETF-Standard syntax to indicate a subpool tunnel. |
| *kbps* | The bandwidth, in kilobits per second, set aside for the MPLS TE tunnel. The range is from 1 to 4294967295. The default value is **0**. |
| *value* | The type of subpool tunnel. The valid entries fo this value are **0** and **1**. |

## Command Default

The default tunnel is a global pool tunnel.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 12.0(5)S | This command was introduced. |
| 12.0(11)ST | The **sub-pool** keyword was added. |
| 12.2(8)T | This command was integrated into Cisco IOS Release 12.2(8)T. |
| 12.2(14)S | This command was integrated into Cisco IOS Release 12.2(14)S. |
| 12.2(28)SB | This command was implemented on the Cisco 10000 (PRE-2) router. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2(33)SRB | The **class-type** keyword was added and the **global** keyword was removed. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |
| Cisco IOS XE Release 3.3S | This command was integrated into Cisco IOS XE Release 3.3S. |
| 15.2(1)T | This command was integrated into Cisco IOS Release 15.2(1)T. |

## Usage Guidelines

Enter the bandwidth for either a global pool (BC0) or a subpool (BC1) tunnel, but not for both in the same statement. To specify both pools, you need to use this command twice, once with the **sub-pool** or **class-type** keyword to specify the narrower tunnel, and once without those keywords to specify the larger tunnel.

## Examples

The following example shows how to configure 100 kbps of bandwidth for the MPLS traffic engineering tunnel:

```
Router(config-if)# tunnel mpls traffic-eng bandwidth 100
```

## Related Commands

| Command | Description |
|---|---|
| **ip rsvp bandwidth** | Enables RSVP for IP on an interface. |
| **show mpls traffic-eng tunnel** | Displays information about tunnels. |

# tunnel mpls traffic-eng exp

To specify the experimental (EXP) bits that will be forwarded over a member tunnel that is part of the Class-Based Tunnel Selection (CBTS) bundle, use the **tunnel mpls traffic-eng exp** command in interface configuration mode. To disable forwarding of the EXP bits, use the **no** form of this command.

**tunnel mpls traffic-eng exp** {*list-of-exp-value*s | **default**}

**no tunnel mpls traffic-eng exp** {*list-of-exp-values*] | **default**}

## Syntax Description

| | |
|---|---|
| *list-of-exp-values* | EXP bits allowed for the interface. Enter up to eight EXP values separated by spaces. Values range from 0 to 7. The default is the EXP values that were not configured or a specific member tunnel. |
| **default** | The member tunnel will forward the packets with the EXP bits that are not being forwarded by other member tunnels that are part of the same bundle. |

## Command Default

No EXP value is assigned to a Multiprotocol Label Switching (MPLS) traffic engineering (TE) tunnel.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.0(29)S | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2(33)SXH | This command was integrated into Cisco IOS Release 12.2(33)SXH. |
| 12.4(20)T | This command was integrated into Cisco IOS Release 12.4(20)T. |

## Usage Guidelines

You should enter the **tunnel mpls traffic-eng exp** command to specify the EXP bits for at least one member tunnel.

With the **tunnel mpls traffic-eng exp** command, you can configure each tunnel with any of the following:

- No EXP-related information
- One or more EXP values for the tunnel to carry (*list-of-exp-values* argument)
- All EXP values not currently allocated to any up tunnel (**default** keyword)
- One or more EXP values for the tunnel to carry, and the property that allows the carrying of all EXP values not currently allocated to any up tunnel (*list-of-exp-values* **default** argument-keyword pair)

The **default** keyword allows you to avoid explicitly listing all possible EXP values. You indicate a preference as to which tunnel to use for certain EXP values, should a tunnel other than the default tunnel go down.

This command allows configurations where:

- Not all EXP values are explicitly allocated to tunnels.
- Multiple tunnels have the default property.
- Some tunnels have EXP values configured and others do not have any configured.
- A given EXP value is configured on multiple tunnels.

The configuration of each tunnel is independent of the configuration of any other tunnel.

## Examples

The following example shows how to specify an EXP value of 5 for MPLS TE tunnel Tunnel1:

```
interface Tunnel1
 tunnel destination 10.0.1.1
 tunnel mpls traffic-eng exp 5
```

## Related Commands

| Command | Description |
|---|---|
| **tunnel mpls traffic-eng exp-bundle master** | Configures a master tunnel. |
| **tunnel mpls traffic-eng exp-bundle member** | Identifies which tunnel is a member (bundled tunnel) of a master tunnel. |

# tunnel mpls traffic-eng exp-bundle master

To configure a master tunnel, use the **tunnel mpls traffic-eng exp bundle master** command in interface configuration mode. To unconfigure a master tunnel, use the **no** form of this command.

**tunnel mpls traffic-eng exp-bundle master**

**no tunnel mpls traffic-eng exp-bundle master**

## Syntax Description

This command has no arguments or keywords.

## Command Default

There is no master tunnel for the bundle.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SRA | This command was introduced. |
| 12.2(33)SXH | This command was integrated into Cisco IOS Release 12.2(33)SXH. |
| 12.4(20)T | This command was integrated into Cisco IOS Release 12.4(20)T. |

## Usage Guidelines

Use the **tunnel mpls traffic-eng exp-bundle master** command to configure a master tunnel. Then specify the **tunnel mpls traffic-eng exp-bundle member** command to identify which tunnels belong to that master tunnel. On the member tunnels, define which experimental (EXP) bit values should be used.

## Examples

The following example specifies that there is a master tunnel that includes tunnels Tunnel20000 through Tunnel20007:

```
interface Tunnel200
 ip unnumbered Loopback0
 ip ospf cost 1
 mpls ip
 tunnel destination 10.10.10.10
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng exp-bundle master
 tunnel mpls traffic-eng exp-bundle member Tunnel20000
 tunnel mpls traffic-eng exp-bundle member Tunnel20001
 tunnel mpls traffic-eng exp-bundle member Tunnel20002
 tunnel mpls traffic-eng exp-bundle member Tunnel20003
 tunnel mpls traffic-eng exp-bundle member Tunnel20004
 tunnel mpls traffic-eng exp-bundle member Tunnel20005
 tunnel mpls traffic-eng exp-bundle member Tunnel20006
 tunnel mpls traffic-eng exp-bundle member Tunnel20007
```

## Related Commands

| Command | Description |
| --- | --- |
| **tunnel mpls traffic-eng exp-bundle member** | Identifies which tunnel is a member (bundled tunnel) of a master tunnel. |

# tunnel mpls traffic-eng exp-bundle member

To identify which tunnel is a member (bundled tunnel) of a master tunnel, use the **tunnel mpls traffic-eng exp-bundle member** command in interface configuration mode. To remove the specified tunnel from being a member of the master tunnel, use the **no** form of this command.

**tunnel mpls traffic-eng exp-bundle member** *tunnel-number*

**no tunnel mpls traffic-eng exp-bundle member** *tunnel-number*

## Syntax Description

| | |
|---|---|
| *tunnel-number* | The tunnel that belongs to a master tunnel. |

## Command Default

The master tunnel has no member tunnels.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SRA | This command was introduced. |
| 12.2(33)SXH | This command was integrated into Cisco IOS Release 12.2(33)SXH. |
| 12.4(20)T | This command was integrated into Cisco IOS Release 12.4(20)T. |

## Usage Guidelines

Enter the **tunnel mpls traffic-eng exp-bundle member** command for each tunnel that you want to be a member of the master tunnel. You should enter this command at least once.

## Examples

The following example specifies that Tunnel1 is a member of the master tunnel:

```
interface Tunnel200
 ip unnumbered Loopback0
 ip ospf cost 1
 mpls ip
 tunnel destination 10.10.10.10
 tunnel mode mpls traffic-eng
tunnel mpls traffic-eng exp-bundle master
 tunnel mpls traffic-eng exp-bundle member Tunnel1
```

## Related Commands

| Command | Description |
|---|---|
| **tunnel mpls traffic-eng exp** | Specifies the EXP bits that will be forwarded over a member tunnel that is part of the CBTS bundle. |
| **tunnel mpls traffic-eng exp-bundle master** | Configures a master tunnel. |

# tunnel mpls traffic-eng fast-reroute

To enable a Multiprotocol Label Switching (MPLS) traffic engineering (TE) tunnel to use an established backup tunnel in the event of a link or node failure, use the **tunnel mpls traffic-eng fast-reroute** command in interface configuration mode. To disable this capability, use the **no** form of this command.

**tunnel mpls traffic-eng fast-reroute** [**bw-protect**] [**node-protect**]

**no tunnel mpls traffic-eng fast-reroute**

## Syntax Description

| | |
|---|---|
| **bw-protect** | (Optional) Sets the "bandwidth protection desired" bit so that backup bandwidth protection is enabled. |
| **node-protect** | (Optional) Sets the "node protection desired" bit so that backup bandwidth protection is enabled. |

## Command Default

There is no backup bandwidth protection.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 12.0(08)ST | This command was introduced. |
| 12.2(18)S | This command was integrated into Cisco IOS Release 12.2(18)S. |
| 12.2(18)SXD | This command was implemented on the Cisco Catalyst 6000 series with the SUP720 processor. |
| 12.0(29)S | The **bw-protect** keyword was added. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.4(20)T | This command was integrated into Cisco IOS Release 12.4(20)T. |
| Cisco IOS XE Release 3.3S | This command was integrated into Cisco IOS XE Release 3.3S. |
| 15.2(1)T | This command was integrated into Cisco IOS Release 15.2(1)T. |

## Usage Guidelines

If you specify the **bw-protect** keyword, all path messages for the tunnel's label switched path (LSP) are sent with the bandwidth protection bit set.

After you enter the command, with or without the **bw-protect** keyword, the requested action or change propagates along all hops of the LSP. Midpoint routers that are the points of local repairs (PLRs) for the LSP take the appropriate action based on whether the bit was just set or cleared. If the bit was just set or cleared, a new backup tunnel selection happens for the LSP because the LSP now has a higher or lower priority in the backup tunnel selection process.

To unconfigure only backup bandwidth protection, enter the **tunnel mpls traffic-eng fast-reroute** command.

To disable an MPLS TE tunnel from using an established backup tunnel in the event of a link or node failure, enter the **no** form of the command.

## Examples

In the following example, backup bandwidth protection is enabled:

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

## Related Commands

| Command | Description |
|---|---|
| **mpls traffic-eng backup-path tunnel** | Configures the interface to use a backup tunnel in the event of a detected failure on the interface. |
| **mpls traffic-eng fast-reroute backup-prot-preemption** | Changes the backup protection preemption algorithm to minimize the amount of bandwidth that is not used. |
| **show tunnel mpls traffic-eng fast-reroute** | Displays information about fast reroute for MPLS traffic engineering. |

# tunnel mpls traffic-eng forwarding-adjacency

To advertise a traffic engineering (TE) tunnel as a link in an Interior Gateway Protocol (IGP) network, use the **tunnel mpls traffic-eng forwarding-adjacency** command in interface configuration mode. To disable the functionality, use the **no** form of this command.

**tunnel mpls traffic-eng forwarding-adjacency** [**holdtime** *milliseconds*]

**no tunnel mpls traffic-eng forwarding-adjacency**

## Syntax Description

| | |
|---|---|
| **holdtime** *milliseconds* | (Optional) Specifies the time, in milliseconds (ms), that a TE tunnel waits after going down before informing the network. The range is 0 to 4294967295 ms. The default value is 0. |

## Command Default

A TE tunnel is not advertised as a link in an IGP network.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.0(15)S | This command was introduced. |
| 12.0(16)ST | This command was integrated into Cisco IOS Release 12.0(16)ST. |
| 12.2(18)S | This command was integrated into Cisco IOS Release 12.2(18)S. |
| 12.2(18)SXD | This command was integrated into Cisco IOS Release 12.2(18)SXD. |
| 12.2(27)SBC | This command was integrated into Cisco IOS Release 12.2(27)SBC. |
| 12.2(28)SB | This command was integrated into Cisco IOS Release 12.2(28)SB. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.4(20)T | This command was integrated into Cisco IOS Release 12.4(20)T. |

## Usage Guidelines

Use the **tunnel mpls traffic-eng forwarding-adjacency** command with the **isis metric** command to avoid inefficient forwarding behavior. Ensure that any nodes traversed by the TE tunnel being advertised do not consider the TE tunnel as part of the shortest path to the destination.

**Note** The **tunnel mpls traffic-eng forwarding-adjacency** command requires Intermediate System-to-Intermediate System (IS-IS) support.

## Examples

In the following example, the holdtime is set to 10,000 milliseconds:

```
Router(config-if)# tunnel mpls traffic-eng forwarding-adjacency holdtime 10000
```

In the following example, the holdtime defaults to 0:

```
Router(config-if)# tunnel mpls traffic-eng forwarding-adjacency
```

## Related Commands

| Command | Description |
|---|---|
| **debug mpls traffic-eng forwarding-adjacency** | Displays debug messages for traffic engineering, forwarding adjacency events. |
| **isis metric** | Configures the cost metric for an interface. |
| **show mpls traffic-eng forwarding-adjacency** | Displays TE tunnels being advertised as links in an IGP network. |

# tunnel mpls traffic-eng interface down delay

To force a tunnel to go down as soon as the headend router detects that the label-switched path (LSP) is down, use the **tunnel mpls traffic-eng interface down delay** command in interface configuration mode. To disable this feature, use the **no** form of this command.

**tunnel mpls traffic-eng interface down delay** *time*

**no tunnel mpls traffic-eng interface down delay** *time*

## Syntax Description

| | |
|---|---|
| *time* | Time, in minutes. The only valid value is 0. |

## Defaults

There is a delay before the tunnel goes down.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.0(30)S | This command was introduced. |
| 12.2(33)SRB | This command was integrated into Cisco IOS Release 12.2(33)SRB. |
| 12.4(20)T | This command was integrated into Cisco IOS Release 12.4(20)T. |

## Usage Guidelines

You cannot specify both the **tunnel mpls traffic-eng interface down delay** command and the **tunnel mpls traffic-eng forwarding-adjacency** command. The first command that you enter would prevent the implementation of the other command and would cause the system to display error messages.

## Examples

In the following example, if the headend router detects that a link has goes down on tunnel 1000, the tunnel goes down immediately.

```
Router(config)# interface tunnel 1000
Router(config-if)# tunnel mpls traffic-eng interface down delay 0
```

# tunnel mpls traffic-eng load-share

To determine load-sharing among two or more Multiprotocol Label Switching (MPLS) traffic engineering (TE) tunnels that begin at the same router and go to an identical destination, use the **tunnel mpls traffic-eng load-share** command in interface configuration mode. To disable this feature, use the **no** form of this command.

**tunnel mpls traffic-eng load-share** *value*

**no tunnel mpls traffic-eng load-share** *value*

## Syntax Description

| | |
|---|---|
| *value* | A value from which the head-end router will calculate the proportion of traffic to be sent down each of the parallel tunnels. Range is from 1 to 1000000. |

## Defaults

No default behavior or values.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.1(3)T | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

Each parallel tunnel must be configured with this command. Specify a value to indicate the *proportion* of total traffic you want to be allocated into each individual tunnel. For example, if there are to be three parallel tunnels, and you want Tunnel1 to carry half of the traffic and the other two tunnels to carry one-quarter, you should enter the following values:

- Tunnel1 — 2
- Tunnel2 — 1
- Tunnel3 — 1

The ability to divide bandwidth in unequal amounts across traffic engineering tunnels has a finite granularity. This granularity varies by platform, with both hardware and software limits. If load-sharing is configured so that it exceeds the available granularity, the following message is displayed:

```
@FIB-4-UNEQUAL: Range of unequal path weightings too large for prefix x.x.x.x/y. Some
available paths may not be used.
```

To eliminate this message, it is recommended that you change the requested bandwidth or loadshare.

## Examples

In the following example, three tunnels are configured, with the first tunnel receiving half of the traffic and the other two tunnels receiving one-quarter:

```
interface Tunnel1
    ip unnumbered Loopback0
    no ip directed-broadcast
    tunnel destination 41.41.41.41
    tunnel mode mpls traffic-eng
    tunnel mpls traffic-eng path-option 10 dynamic
    tunnel mpls traffic-eng load-share 2

interface Tunnel2
    ip unnumbered Loopback0
    no ip directed-broadcast
    tunnel destination 41.41.41.41
    tunnel mode mpls traffic-eng
    tunnel mpls traffic-eng path-option 10 dynamic
    tunnel mpls traffic-eng load-share 1

interface Tunnel3
    ip unnumbered Loopback0
    no ip directed-broadcast
    tunnel destination 41.41.41.41
    tunnel mode mpls traffic-eng
    tunnel mpls traffic-eng path-option 10 dynamic
    tunnel mpls traffic-eng load-share 1
```

## Related Commands

| Command | Description |
|---|---|
| **show ip route** | Displays routing table information about tunnels, including their traffic share. |
| **tunnel mpls traffic-eng bandwidth** | Configures bandwidth in Kbps for an MPLS traffic engineering tunnel. |

# tunnel mpls traffic-eng name

To provide a name for a Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) point-to-point (P2P) or point-to-multipoint (P2MP) tunnel, use the **tunnel mpls traffic-eng name** command in tunnel interface configuration mode. To remove the name from the tunnel, use the **no** form of this command.

**tunnel mpls traffic-eng name** *signaled-tunnel-name*

**no tunnel mpls traffic-eng name** *signaled-tunnel-name*

## Syntax Description

| | |
|---|---|
| *signaled-tunnel-name* | Name of the tunnel.<br>Limit: 63 characters<br>Spaces are not allowed. |

## Command Default

The TE tunnel name is either the interface description or is *hostname*_t*tunnel id*.

## Command Modes

Tunnel interface configuration mode

## Command History

| Release | Modification |
|---|---|
| 15.1(1)S | This command was introduced. |

## Usage Guidelines

When configuring the tunnel name, consider the following:

- If tunnel name is configured, it overrides the default names, which are either the tunnel interface description or *hostname*_t*tunnel id*. If the TE tunnel name configuration is removed, TE resignals the LSP using the next preferred tunnel name source (the interface description or the default host name and tunnel ID). This is completed in break-before-make fashion; therefore, traffic may be lost.
- The TE tunnel name must be unique. It cannot be the same name as the interface description or the hostname and tunnel id.
- The command is available for tunnels that are configured in TE P2P tunnel mode or TE P2MP tunnel mode.
- In releases previous to Cisco IOS 15.1(1)S, changing the interface description does NOT result in the LSP being resignaled. The introduction of the **tunnel mpls traffic-eng name** command requires that the tunnel state be flapped before the signaled name is updated.

## Examples

The following example specifies the name of tunnel0 as "MYTUNNEL" and tunnel1 as "MYOTHERTUNNEL":

```
Router(config)# interface tunnel0

Router(config-if)# tunnel mpls traffic-eng name MYTUNNEL
.
.
.
```

```
Router(config)# interface tunnel1

Router(config-if)# tunnel mpls traffic-eng name MYOTHERTUNNEL
```

The **show mpls traffic-eng tunnel** commands display the names of the P2P and P2MP tunnels.

```
Router# show mpls traffic-eng tunnel tunnel0
Name: MYTUNNEL                      (Tunnel0) Destination: 10.3.0.1

Router# show mpls traffic-eng tunnel tunnel1

Tunnel1     (p2mp),  Admin: up, Oper: up
  Name: MYOTHERTUNNEL
```

The **show mpls traffic-eng tunnel brief**' command displays the name of P2P tunnels, However, for P2MP tunnels, the command displays the tunnel ID and not the name. In the following example, the output displays the name of the P2P tunnel0 and the tunnel ID of P2MP tunnel1.

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

P2P TUNNELS/LSPs:
TUNNEL NAME            DESTINATION      UP IF     DOWN IF   STATE/PROT
MYTUNNEL               10.3.0.1         -         Et0/0     up/up
Displayed 1 (of 1) heads, 0 (of 0) midpoints, 0 (of 0) tails

P2MP TUNNELS:
                         DEST        CURRENT
INTERFACE   STATE/PROT  UP/CFG     TUNID  LSPID
Tunnel1     up/up       2/3        1      1
Displayed 1 (of 1) P2MP heads
```

## Related Commands

| Command | Description |
|---|---|
| **show mpls traffic-eng tunnel** | Displays information about the MPLS Traffic Engineering P2P or P2MP tunnels. |

# tunnel mpls traffic-eng path-option

To configure a path option for a Multiprotocol Label Switching (MPLS) traffic engineering (TE) tunnel, use the **tunnel mpls traffic-eng path-option** command in interface configuration mode. To disable this function, use the **no** form of this command.

**tunnel mpls traffic-eng path-option** {*number* {**dynamic** [**attributes** *lsp-attributes* | **bandwidth** {*kbps* | **subpool** *kbps*} [**lockdown**] | **lockdown** [**bandwidth** {*kbps* | **subpool** *kbps*}] | **explicit** {**identifier** *path-number* | **name** *path-name*} [**attributes** *lsp-attributes* [**verbatim**]] | **bandwidth** {*kbps* | **subpool** *kbps*} [**lockdown**] [**verbatim**]] | **lockdown bandwidth** {*kbps* | **subpool** *kbps*} [**verbatim**] | **verbatim bandwidth** {*kbps* | **subpool** *kbps*} [**lockdown**]}}

**no tunnel mpls traffic-eng path-option** *number*

## Syntax Description

| | |
|---|---|
| *number* | Preference for this path option. When you configure multiple path options, lower numbered options are preferred. Valid values are from 1 to 1000. |
| **dynamic** | Dynamically calculates the path of the label switched path (LSP) |
| **attributes** *lsp-attributes* | (Optional) Identifies an LSP attribute list. **Note** The attribute list used should be the same as the primary path option being configured. |
| **bandwidth** *kbps* | (Optional) Overrides the bandwidth configured on the tunnel or the attribute list. The *kbps* is the number of kilobits per second set aside for the path option. The range is from 1 to 4294967295. **Note** The bandwidth value should be the same as the primary path option being configured. |
| **subpool** *kbps* | (Optional) Indicates that the bandwidth override value uses the subpool bandwidth. The *kbps* argument is the number of kilobits per second of the subpool bandwidth set aside for the path option. The range is from 1 to 4294967295. |
| **lockdown** | (Optional) Indicates that the LSP cannot be reoptimized. |
| **explicit** | Specifies that the path of the LSP is an IP explicit path. |
| **identifier** *path-number* | Specifies the path number of the IP explicit path that the tunnel uses with this option. The range is from 1 to 65535. |
| **name** *path-name* | Specifies the path name of the IP explicit path that the tunnel uses with this option. |
| **verbatim** | (Optional) Bypasses the topology database verification process. |

## Command Default

No path option for an MPLS TE tunnel is configured.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 12.2(4)T | This command was introduced. |
| 12.4(20)T | This command was integrated into Cisco IOS Release 12.4(20)T. |
| Cisco IOS XE Release 3.3S | This command was integrated into Cisco IOS XE Release 3.3S. |
| 15.2(1)T | This command was integrated into Cisco IOS Release 15.2(1)T. |

## Usage Guidelines

You can configure multiple path options for a single tunnel. For example, there can be several explicit path options and a dynamic option for one tunnel. Path setup preference is for lower (not higher) numbers, so option 1 is preferred.

If you specify the **dynamic** keyword, the software checks both the physical bandwidth of the interface and the available TE bandwidth to be sure that the requested amount of bandwidth does not exceed the physical bandwidth of any link. To oversubscribe links, you must specify the **explicit** keyword. If you use the **explicit** keyword, the software only checks how much bandwidth is available on the link for TE; the amount of bandwidth you configure is not limited to how much physical bandwidth is available on the link.

## Examples

The following example shows how to configure the tunnel to use a named IP explicit path:

```
Router(config-if)# tunnel mpls traffic-eng path-option 1 explicit name test
```

## Related Commands

| Command | Description |
|---|---|
| **ip explicit-path** | Enters the command mode for IP explicit paths and creates or modifies the specified path. |
| **mpls traffic-eng lsp attributes** | Creates or modifies an LSP attribute list. |
| **show ip explicit-paths** | Displays the configured IP explicit paths. |
| **tunnel mpls traffic-eng path-option protect** | Configures a secondary path option for an MPLS TE tunnel. |

# tunnel mpls traffic-eng path-option protect

To configure a secondary path option for a Multiprotocol Label Switching (MPLS) traffic engineering (TE) tunnel, use the **tunnel mpls traffic-eng path-option protect** command in interface configuration mode. To disable this function, use the **no** form of this command.

### Cisco IOS Release 12.0(30)S and Later

**tunnel mpls traffic-eng path-option protect** *number* [**attributes** *lsp-attributes* | **bandwidth** {*kbps* | **sub-pool** *kbps*} | **explicit** {**identifier** *path-number* | **name** *path-name*} [**attributes** *lsp-attributes* [**verbatim**] | **bandwidth** {*kbps* | **sub-pool** *kbps*} [**verbatim**] | **verbatim** [**bandwidth** {*kbps* | **sub-pool** *kbps*}]] | **list** {**identifier** *path-number* | **name** *path-name*} [**attributes** *lsp-attributes* | **bandwidth** {*kbps* | **sub-pool** *kbps*}]]

### Cisco IOS Release 12.4(20)T and Later

**tunnel mpls traffic-eng path-option protect** *number* {**dynamic** [**attributes** *lsp-attributes* | **bandwidth** {*kbps* | **sub-pool** *kbps*}] | **explicit** {**identifier** *path-number* | **name** *path-name*} [**attributes** *lsp-attributes* [**verbatim**] | **bandwidth** {*kbps* | **sub-pool** *kbps*} [**verbatim**] | **verbatim** [**bandwidth** {*kbps* | **sub-pool** *kbps*}]]}

### Cisco IOS Release 12.2(50)SY and Later

**tunnel mpls traffic-eng path-option protect** *number* **explicit** {**identifier** *path-number* | **name** *path-name*} [**attributes** *lsp-attributes* [**verbatim**] | **bandwidth** {*kbps* | **sub-pool** *kbps*} [**verbatim**] | **verbatim** [**bandwidth** {*kbps* | **sub-pool** *kbps*}]]

**no tunnel mpls traffic-eng path-option protect** *number*

## Syntax Description

| | |
|---|---|
| *number* | The primary path option being protected. Valid values are from 1 to 1000. |
| **dynamic** | Dynamically calculated part of the label switched path (LSP). |
| **attributes** *lsp-attributes* | (Optional) Identifies an LSP attribute list.<br>**Note** The attribute list used should be the same as the primary path option being protected. |
| **bandwidth** *kbps* | (Optional) Overrides the bandwidth configured on the tunnel or the attribute list. The value *kbps* is the number of kilobits per second set aside for the path option. The range is from 1 to 4294967295.<br>**Note** The bandwidth value should be the same as the primary path option being protected. |
| **sub-pool** *kbps* | (Optional) Indicates that the bandwidth override value uses the sub-pool bandwidth. The value *kbps* is the number of kilobits per second of the sub-pool bandwidth set aside for the path option. The range is from 1 to 4294967295. |
| **explicit** | Indicates that the path of the LSP is an IP-explicit path. |
| **identifier** *path-number* | Specifies the path number of the IP-explicit path that the tunnel uses with this option. The range is from 1 to 65535. |

| | |
|---|---|
| **name** *path-name* | Specifies the path name of the IP-explicit path that the tunnel uses with this option. |
| **verbatim** | (Optional) Bypasses the topology database verification process. |

## Command Default

The MPLS TE tunnel does not have a secondary path option.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 12.0(5)S | This command was introduced. |
| 12.0(26)S | This command was modified. LSP-related keywords and arguments for path options were added. |
| 12.0(30)S | This command was modified. The **protect** keyword was added. |
| 12.2(33)SRC | This command was integrated into Cisco IOS Release 12.2(33)SRC. |
| 12.4(20)T | This command was integrated into Cisco IOS Release 12.4(20)T. The **dynamic** keyword was added. |
| 12.2(50)SY | This command was integrated into Cisco IOS Release 12.2(50)SY. The dynamic keyword is not available on Cisco Catalyst 6000 platforms. |

## Usage Guidelines

Cisco recommends that the primary path options that are being protected use explicit paths.

Calculation of a dynamic path for the path-protected LSP is not available. When configuring the IP explicit path for the path-protected LSP, choose hops that minimize the number of links and nodes shared with the protected primary path option.

If the path option being protected uses an attribute list, configure path protection to use the same attribute list.

If the path option being protected uses bandwidth override, configure path protection to use bandwidth override with the same values.

## Examples

The following example shows how to configure the tunnel to use a named IP-explicit path:

```
Router(config-if)# tunnel mpls traffic-eng path-option protect 1 explicit name test
```

The following example shows how to configure path option 1 to use an LSP attribute list identified with the numeral 1:

```
Router(config-if)# tunnel mpls traffic-eng path-option protect 1 explicit name test
attributes 1
```

The following example shows how to configure bandwidth for a path option to override the bandwidth configured on the tunnel:

```
Router(config-if)# tunnel mpls traffic-eng path-option protect 3 explicit name test
bandwidth 0
```

The following example shows how to configure path protection on a standby LSP:

```
Router(config-if)# tunnel mpls traffic-eng path-option 10 explicit name pri-path
Router(config-if)# tunnel mpls traffic-eng path-option protect 10 explicit name alt-path
```

## Related Commands

| Command | Description |
|---|---|
| **ip explicit-path** | Enters the command mode for IP-explicit paths and creates or modifies the specified path. |
| **mpls traffic-eng lsp attributes** | Creates or modifies an LSP attribute list. |
| **show ip explicit-paths** | Displays the configured IP-explicit paths. |
| **tunnel mpls traffic-eng path-option** | Configures a primary path for an MPLS TE tunnel. |

# tunnel mpls traffic-eng path-selection metric

To specify the metric type to use for path calculation for a tunnel, use the **tunnel mpls traffic-eng path-selection metric** command in interface configuration mode. To remove the specified metric type, use the **no** form of this command.

**tunnel mpls traffic-eng path-selection metric** {**igp** | **te**}

**no tunnel mpls traffic-eng path-selection metric**

## Syntax Description

| | |
|---|---|
| **igp** | Use the Interior Gateway Protocol (IGP) metric. |
| **te** | Use the traffic engineering (TE) metric. |

## Defaults

The default is the **te** metric.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.0(18)ST | This command was introduced. |
| 12.2(14)S | This command was integrated into Cisco IOS Release 12.2(14)S. |
| 12.2(28)SB | This command was integrated into Cisco IOS Release 12.2(28)SB. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |
| 12.4(20)T | This command was integrated into Cisco IOS Release 12.4(20)T. |

## Usage Guidelines

The metric type to be used for path calculation for a given tunnel is determined as follows:

- If the **tunnel mpls traffic-eng path-selection metric** command was entered to specify a metric type for the tunnel, use that metric type.
- Otherwise, if the **mpls traffic-eng path-selection metric** was entered to specify a metric type, use that metric type.
- Otherwise, use the default (**te**) metric.

## Examples

The following commands specify that the igp metric should be used when you are calculating the path for Tunnel102:

```
Router(config)# interface tunnel102
Router(config-if)# tunnel mpls traffic-eng path-selection metric igp
```

## Related Commands

| Command | Description |
|---|---|
| **mpls traffic-eng path-selection metric** | Specifies the metric type to use for path calculation for TE tunnels for which no metric has been explicitly configured. |

# tunnel mpls traffic-eng priority

To configure the setup and reservation priority for Multiprotocol Label Switching (MPLS) traffic engineering (TE) tunnel, use the **tunnel mpls traffic-eng priority** command in interface configuration mode. To remove the specified setup and reservation priority, use the **no** form of this command.

**tunnel mpls traffic-eng priority** *setup-priority* [*hold-priority*]

**no tunnel mpls traffic-eng priority** *setup-priority* [*hold-priority*]

## Syntax Description

| | |
|---|---|
| *setup-priority* | The priority used when signaling an link-state packet (LSP) for this tunnel to determine which existing tunnels can be preempted. Valid values are from 0 to 7, where a lower number indicates a higher priority. Therefore, an LSP with a setup priority of 0 can preempt any LSP with a non-0 priority. |
| *hold-priority* | (Optional) The priority associated with an LSP for this tunnel to determine if it should be preempted by other LSPs that are being signaled. Valid values are from 0 to 7, where a lower number indicates a higher priority. |

## Command Default

By default, the setup priority is 7. The value of hold priority is the same as the value of setup priority.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 12.0(5)S | This command was introduced. |
| 12.2(28)SB | This command was integrated into Cisco IOS Release 12.2(28)SB. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |
| Cisco IOS XE Release 3.3S | This command was integrated into Cisco IOS XE Release 3.3S. |

## Usage Guidelines

When an LSP is being signaled and an interface does not currently have enough bandwidth available for that LSP, the call admission software preempts lower-priority LSPs so that the new LSP can be admitted. (LSPs are preempted if that allows the new LSP to be admitted.)

The new LSP's priority is its setup priority and the existing LSP's priority is its hold priority. The two priorities enables the signaling of an LSP with a low setup priority (so that the LSP does not preempt other LSPs on setup) but a high hold priority (so that the LSP is not preempted after it is established).

Setup priority and hold priority are typically configured to be equal, and setup priority cannot be better (numerically smaller) than the hold priority.

## Examples

The following example shows how to configure a tunnel with a setup and hold priority of 1:

```
Router(config-if)# tunnel mpls traffic-eng priority 1 1
```

## Related Commands

| Command | Description |
|---|---|
| **tunnel mode mpls traffic-eng** | Sets the mode of a tunnel to MPLS for traffic engineering. |

# tunnel mpls traffic-eng record-route

To include the interface address for the label switched path (LSP) in the Record Route Object (RRO) for an RESV message, use the **tunnel mpls traffic-eng record-route** command in interface configuration mode. To remove the interface address for the LSP in the RRO for the RESV message, use the **no** form of this command.

**tunnel mpls traffic-eng record-route**

**no tunnel mpls traffic-eng record-route**

## Syntax Description

This command has no arguments or keywords.

## Command Default

By default, this command is disabled. The interface addresses for the LSP are not included in the RRO of the RESVmessage.The **record-route** option is automatically enabled when the **tunnel mpls traffic-eng fast-reroute** command for the fast-reroute (FRR) feature is enabled at the headend.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.0(5)S | This command was introduced. |
| 12.0(7)T | This command was integrated into Cisco IOS Release 12.0(7)T. |

## Usage Guidelines

The RRO has two functions. It records the route of the LSP that can be used in loop prevention, and it records labels that are used by FRR.

The contents of a RRO are a series of variable-length data items called subobjects.

If record route is enabled, the RRO contains details in the following order: node-ID, interface address, and label.

## Examples

The following example shows how to include the interface address using the **tunnel mpls traffic-eng record-route** command:

```
interface tunnel1
ip unnumbered loopback0
no ip direct-broadcast
tunnel destination 192.168.1.5
tunnel mode mpls traffic-eng
tunnel mpls traffic-eng autoroute announce
tunnel mpls traffic-eng bandwidth 100
tunnel mpls traffic-eng priority 1 1
tunnel mpls traffic-eng path-option 1 dynamic
tunnel mpls traffic-eng record-route
```

**Related Commands,**

| Command | Description |
|---|---|
| **show ip rsvp reservation** | Displays current RSVP related receiver information in the database. |
| **show mpls traffic-eng tunnels** | Displays information on the source, destination, path and interface of MPLS TE tunnels. |
| **tunnel mpls traffic-eng fast-reroute** | Enables an MPLS TE tunnel to use an established backup tunnel in the event of a link or node failure. |

# tunnel tsp-hop

To define hops in the path for the label switching tunnel, use the **tunnel tsp-hop** command in interface configuration mode. To remove these hops, use the **no** form of this command.

**tunnel tsp-hop** *hop-number ip-address* [**lasthop**]

**no tunnel tsp-hop** *hop-number ip-address* [**lasthop**]

## Syntax Description

| | |
|---|---|
| *hop-number* | The sequence number of the hop being defined in the path. The first number is 1, which identifies the hop just after the head hop. |
| *ip-address* | The IP address of the input interface on that hop. |
| **lasthop** | (Optional) Indicates that the hop being defined is the final hop in the path (the tunnel destination). |

## Defaults

No hops are defined.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 11.1 CT | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

The list of tunnel hops must specify a strict source route for the tunnel. In other words, the router at hop <*n*> must be directly connected to the router at hop <*n*>+1.

## Examples

The following example shows how to configure a two-hop tunnel. The first hop router/switch is 172.16.0.2, and the second and last hop is router/switch 172.17.0.2.

```
Router(config)# interface tunnel 5

Router(config-if)# tunnel mode mpls traffic-eng
Router(config-if)# ip unnumbered e0/1
Router(config-if)# tunnel tsp-hop 1 172.16.0.2
Router(config-if)# tunnel tsp-hop 2 172.17.0.2 lasthop
```

## Related Commands

| Command | Description |
|---|---|
| **tunnel mpls traffic-eng affinity** | Sets the encapsulation mode of the tunnel to label switching. |

# udp port

To configure the User Datagram Protocol (UDP) port information on the xconnect class, use the **udp port** command in xconnect configuration mode. To revert to the default settings, use the **no** form of this command.

**udp port local** *local-udp-port* **remote** *remote-udp-port*

**no udp port local** *local-udp-port* **remote** *remote-udp-port*

## Syntax Description

| | |
|---|---|
| **local** *local-udp-port* | The local UDP port number. Valid values are from 49152 to 57343. |
| **remote** *remote-udp-port* | Specifies the remote UDP port number. Valid values are from 49152 to 57343. |

## Command Default

The virtual cicuit will not be enabled.

## Command Modes

Xconnect configuration mode (config-if-xconn)

## Command History

| Release | Modification |
|---|---|
| 15.1(2)S | This command was introduced. |

## Examples

The following example shows how to configure the local and remote UDP port numbers:

```
Router# configure terminal
Router(config)# interface cem 0/13
Router(config-if)# xconnect 10.2.2.9 200 pw-class udpClass
Router(config-if-xconn)# udp port local 50000 remote 57343
```

## Related Commands

| Command | Description |
|---|---|
| **encapsulation (pseudowire)** | Specifies an encapsulation type for tunneling Layer 2 traffic over a pseudowire. |
| **show pw-udp vc** | Displays information about pseudowire UDP VC. |
| **xconnect** | Binds an attachment circuit to an L2TPv3 pseudowire for xconnect service and enters xconnect configuration mode. |

# vpn id

To set or update a Virtual Private Network (VPN) ID on a VPN routing and forwarding (VRF) instance, use the **vpn id** command in VRF configuration mode.

**vpn id** *oui*:*vpn-index*

## Syntax Description

| | |
|---|---|
| *oui*: | Organizationally unique identifier (OUI). The IEEE organization assigns this identifier to companies. The OUI is restricted to three octets and followed by a colon. |
| *vpn-index* | Identifies the VPN within the company. This VPN index is restricted to four octets. |

## Defaults

The VPN ID is not set.

## Command Modes

VRF configuration

## Command History

| Release | Modification |
|---|---|
| 12.0(17)ST | This command was introduced. |
| 12.2(4)B | This command was integrated into Cisco IOS Release 12.2(4)B. |
| 12.2(8)T | This command was integrated into Cisco IOS Release 12.2(8)T. |
| 12.2(14)S | This command was integrated into Cisco IOS Release 12.2(14)S. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

Each VRF configured in a provider edge (PE) router can have a VPN ID. Use the same VPN ID for the PE routers that belong to the same VPN. Make sure the VPN ID is unique for each VPN in the service provider network.

Once configured, a VPN ID cannot be removed, however, it can be changed. To change the VPN ID, issue the command again. The new ID overwrites the existing ID.

## Examples

The following example shows how to assign the VPN ID of 0000a100003f6c to a VRF called vpn1:

```
Router(config)# ip vrf vpn1
Router(config-vrf)# vpn id a1:3f6c
```

## Related Commands

| Command | Description |
| --- | --- |
| **show ip vrf detail** | Displays all the VRFs on a router. |
| **show ip vrf id** | Displays all the VPN IDs that are configured in the router and their associated VRF names and VRF RDs. |

# vrf definition

To configure a Virtual Private Network (VPN) routing and forwarding (VRF) routing table instance and enter VRF configuration mode, use the **vrf definition** command in global configuration mode. To remove a VRF routing table, use the **no** form of this command.

**vrf definition** *vrf-name*

**no vrf definition** *vrf-name*

## Syntax Description

| | |
|---|---|
| *vrf-name* | Name assigned to a VRF. |

## Command Default

No VRFs are defined.
No import or export lists are associated with a VRF.
No route maps are associated with a VRF.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SRB | This command was introduced. |
| 12.2(33)SXH | This command was integrated into Cisco IOS Release 12.2(33)SXH. |
| 12.2(33)SB | This command was integrated into Cisco IOS Release 12.2(33)SB. |
| 12.4(20)T | This command was integrated into Cisco IOS Release 12.4(20)T. |
| 12.2(33)SXI | This command was integrated into Cisco IOS Release 12.2(33)SXI. |

## Usage Guidelines

Use the **vrf definition** command to give a VRF a name and to enter VRF configuration mode. Once the router is in VRF configuration mode, use the **rd** command to give the VRF a route distinguisher (RD). The **rd** command creates the routing and forwarding tables and associates the RD with the VRF instance named in the *vrf-name* argument.

Users can configure shared route targets (import and export) between IPv4 and IPv6. This feature is useful in a migration scenario, where IPv4 policies already are configured and IPv6 policies should be the same as the IPv4 policies. You can configure separate route-target policies for IPv4 and IPv6 VPNs in address family configuration mode. Enter address family configuration mode from VRF configuration mode.

In VRF configuration mode, you can also associate a Simple Network Management Protocol (SNMP) context with the named VRF and configure or update a VPN ID.

The **vrf definition default** command can be used to configure a VRF name that is a NULL value until a default VRF name can be configured. This is typically before any VRF related AAA commands are configured.

## Examples

The following example assigns the name vrf1 to a VRF, enters VRF configuration mode, and configures a route distinguisher, 100:20:

```
Router(config)# vrf definition vrf1
Router(config-vrf)# rd 100:20
```

## Related Commands

| Command | Description |
|---|---|
| **address-family** | Enters address family configuration mode for configuring routing protocols such as BGP, RIP, and static routing. |
| **context** | Associates an SNMP context with a particular VRF. |
| **rd** | Specifies a route distinguisher. |
| **route-target** | Creates a route-target extended community for a VPN VRF. |
| **vpn id** | Sets or updates a VPN ID on a VRF. |
| **vrf forwarding** | Associates a VRF instance with an interface or subinterface. |

# vrf forwarding

To associate a Virtual Private Network (VPN) routing and forwarding (VRF) instance with an interface or subinterface, use the **vrf forwarding** command in interface configuration mode. To disassociate a VRF from an interface, use the **no** form of this command.

**vrf forwarding** *vrf-name*

**no vrf forwarding** *vrf-name*

## Syntax Description

| | |
|---|---|
| *vrf-name* | Name assigned to a VRF. |

## Command Default

The default for an interface is the global routing table.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SRB | This command was introduced. |
| 12.2(33)SXH | This command was integrated into Cisco IOS Release 12.2(33)SXH. |
| 12.2(33)SB | This command was integrated into Cisco IOS Release 12.2(33)SB. |
| 12.4(20)T | This command was integrated into Cisco IOS Release 12.4(20)T. |
| 12.2(33)SXI | This command was integrated into Cisco IOS Release 12.2(33)SXI. |

## Usage Guidelines

Use the **vrf forwarding** command to associate an interface with a VRF. When the interface is bound to a VRF, previously configured IPv4 and IPv6 addresses are removed, and they must be reconfigured.

## Examples

The following example shows how to associate a VRF named site1 to serial interface 0/0 and configure an IPv6 and an IPv4 address:

```
interface Serial0/0
 vrf forwarding site1
 ipv6 address 2001:100:1:1000::72b/64
 ip address 10.11.11.1 255.255.255.0
```

## Related Commands

| Command | Description |
|---|---|
| **vrf definition** | Configures a VRF routing table instance and enters VRF configuration mode. |

# vrf selection source

To populate a single source IP address, or range of source IP addresses, to a VRF Selection table, use the **vrf selection source** command in global configuration mode. To remove a single source IP address or range of source IP addresses from a VRF Selection table, use the **no** form of this command.

**vrf selection source** *source-IP-address source-IP-mask* **vrf** *vrf-name*

**no vrf selection source** *source-IP-address source-IP-mask* **vrf** *vrf-name*

## Syntax Description

| | |
|---|---|
| *source-IP-address* | New source IP address to be added to the VRF Selection table. |
| *source-IP-mask* | IP mask for the source IP address or range of single source IP addresses to be added to the VRF Selection table. |
| **vrf** *vrf-name* | Name of the VRF Selection table to which the single source IP address or range of source IP addresses should be added. |

## Defaults

No default behavior or values.

## Command Modes

Global configuration

## Command History

| Release | Modification |
|---|---|
| 12.0(22)S | This command was introduced. |
| 12.0(23)S | This command was integrated into Cisco IOS Release 12.0(23)S. |
| 12.0(24)S | This command was integrated into Cisco IOS Release 12.0(24)S. |
| 12.2(14)SZ | This command was integrated into Cisco IOS Release 12.2(14)SZ to support the Cisco 7304 router. |
| 12.2(18)S | This command was integrated into Cisco IOS Release 12.2(18)S to support the Cisco 7304 router. |
| 12.0(26)S | This command was integrated into Cisco IOS Release 12.0(26)S to support the Cisco 7200 and 7500 series routers. |
| 12.2(25)S | This command was integrated into Cisco IOS Release 12.2(25)S to support the Cisco 7200 and 7500 series routers. |
| 12.2(27)SBC | This command was integrated into Cisco IOS Release 12.2(27)SBC. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

If a VRF table is removed by using the **no ip vrf** *vrf-name* command in global configuration mode, all configurations associated with that VRF will be removed including those configurations added with the **vrf selection source** command.

## Examples

The following example shows how to populate the VRF Selection table vpn1 with a source IP network address 10.0.0.0 and the IP mask 255.0.0.0, which would forward any packets with the source IP address 10.0.0.0 into the VRF instance vpn1:

```
Router(config)# vrf selection source 10.0.0.0 255.0.0.0 vrf vpn1
```

The following example shows the message you receive after you have removed the source IP network address 107.1.1.1 and the IP mask 255.255.255.255 from the VRF Selection table vpn1:

```
Router (config)# no vrf selection source 10.1.1.1 255.255.255.255 vrf vpn1
Router (config)#
VRF Selection Configuration: addr:10.1.1.1, mask:255.255.255.255, vrf_name:vpn1

5d13h: VRF Selection Remove Configuration: addr:10.1.1.1, mask: 255.255.255.255
Router (config)#
```

The following example shows the message you receive after you have added the source IP network address 10.1.1.1 and the IP mask 255.255.255.255 to the VRF Selection table vpn1:

```
Router (config)# vrf selection source 10.1.1.1 255.255.255.255 vrf vpn1
Router (config)#
VRF Selection Configuration: addr:10.1.1.1, mask:255.255.255.255, vrf_name:vpn1
VRF Selection: VRF table vpn1, id is: 1
```

## Related Commands

| Command | Description |
|---|---|
| **ip vrf receive** | Adds all the IP addresses that are associated with an interface into a VRF table. |
| **ip vrf select source** | Enables VRF Selection on an interface. |

# vrf upgrade-cli

To upgrade a Virtual Private Network (VPN) routing and forwarding (VRF) instance or all VRFs on the router to support multiple address families (multi-AFs) for the same VRF, use the **vrf upgrade-cli** command in global configuration mode.

**vrf upgrade-cli multi-af-mode** {**common-policies** | **non-common-policies**} [**vrf** *vrf-name*] [**force**]

## Syntax Description

| | |
|---|---|
| **multi-af-mode** | Specifies an upgrade of a single-protocol VRF or all VRFs to a multiprotocol VRF that supports multi-AFs configuration. |
| **common-policies** | Specifies to copy the route-target policies to the common part of the VRF configuration so that the policies apply to all address families configured in the multi-AF VRF. |
| **non-common-policies** | Specifies to copy the route-target policies to the IPv4 address family part of the VRF configuration so that the policies apply only to an IPv4 VRF. |
| **vrf** | (Optional) Specifies a VRF for the upgrade to a multi-AF VRF configuration. |
| *vrf-name* | (Optional) The name of the single-protocol VRF to upgrade to a multi-AF VRF configuration. |
| **force** | (Optional) Disables the prompt to confirm the upgrade process. |

## Command Default

If you do not enter the name of a specific single-protocol VRF, all VRFs defined on the router are upgraded to the multi-AF VRF configuration.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SRB | This command was introduced. |
| 12.2(33)SB | This command was integrated into Cisco IOS Release 12.2(33)SB. |
| 12.4(20)T | This command was integrated into Cisco IOS Release 12.4(20)T. |
| 12.2(33)SXI | This command was integrated into Cisco IOS Release 12.2(33)SXI. |
| 12.2(33)SRE2 | This command was modified to add the **force** keyword. |

## Usage Guidelines

The **vrf upgrade-cli** command is used to upgrade a specified single-protocol VRF (IPv4-only VRF) configuration or all single-protocol VRF configurations on the router to a multiprotocol VRF that supports multi-AF configuration.

The upgrade is automatic and does not require any further configuration. After you enter the **vrf upgrade-cli** command, the single-protocol VRF configuration is lost when you save the configuration to NVRAM. A multiprotocol VRF configuration is saved.

If your configuration requires that all route-target policies (import, export, both) apply to all address families, you enter the **vrf upgrade-cli multi-af-mode common-policies** command. If your configuration requires that these policies apply to IPv4 VPNs only, enter the **vrf upgrade-cli multi-af-mode non-common-policies** command.

After the upgrade to a multiprotocol VRF is complete, you can edit the VRF only with multiprotocol VRF configuration commands.

## Examples

The following example shows how to upgrade a single-protocol VRF configuration named vrf1 to a multi-AF VRF configuration and apply the common policies of vrf1 to all address families defined for the VRF:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
!
Router(config)# vrf upgrade-cli multi-af-mode common-policies vrf vrf1

You are about to upgrade to the multi-AF VRF syntax commands.
You will lose any IPv6 address configured on interfaces
belonging to upgraded VRFs.
Are you sure ? [yes]: yes
Number of VRFs upgraded: 1
Router(config)# exit
```

The following example is a duplicate of the previous upgrade example, but specifies the **force** keyword to disable upgrade confirmation prompts and warnings:

```
Router(config)# vrf upgrade-cli multi-af-mode common-policies vrf vrf1

Number of VRFs upgraded: 1
Router(config)# exit
```

The following is an example of the single-protocol VRF configuration for VRF vrf1 before you enter the **vrf upgrade-cli** command to upgrade to a multi-AF multiprotocol VRF configuration:

```
!
ip vrf vrf1
 rd 1:1
 route-target export 1:1
 route-target import 1:1

interface Loopback1
 ip vrf forwarding vrf1
 ip address 10.3.3.3 255.255.255.255
```

This is an example of the multi-AF multiprotocol VRF configuration for VRF vrf1 after you enter the **vrf upgrade-cli** command with the **common-policies** keyword:

```
!
vrf definition vrf1
 rd 1:1
 route-target export 1:1
 route-target import 1:1
 !
 address-family ipv4
 exit-address-family
!

interface Loopback1
 vrf forwarding vrf1
 ip address 10.3.3.3 255.255.255.255
```

## Related Commands

| Command | Description |
|---|---|
| **show vrf** | Displays the defined VRF instances. |
| **vrf definition** | Configures a VRF routing table instance and enters VRF configuration mode. |
| **vrf forwarding** | Associates a VRF instance with an interface or subinterface. |

# xconnect

To bind an attachment circuit to a pseudowire, and to configure an Any Transport over MPLS (AToM) static pseudowire, use the **xconnect** command in one of the supported configuration modes. To restore the default values, use the **no** form of this command.

**xconnect** *peer-ip-address vc-id* {**encapsulation** {**l2tpv3** [**manual**] | **mpls** [**manual**]} | **pw-class** *pw-class-name* }[**pw-class** *pw-class-name*] [**sequencing** {**transmit** | **receive** | **both**}]

**no xconnect**

### Cisco uBR10012 Router and Cisco uBR7200 Series Universal Broadband Routers

**xconnect** *peer-ip-address vc-id* **encapsulation mpls** [*pw-type*]

**no xconnect** *peer-ip-address vc-id* **encapsulation mpls** [*pw-type*]

## Syntax Description

| | |
|---|---|
| *peer-ip-address* | IP address of the remote provider edge (PE) peer. The remote router ID can be any IP address, as long as it is reachable. |
| *vc-id* | The 32-bit identifier of the virtual circuit (VC) between the PE routers. |
| **encapsulation** {**l2tpv3** [**manual**] \| **mpls** [**manual**]} | Specifies the tunneling method to encapsulate the data in the pseudowire:<br>• **l2tpv3**—Specifies Layer 2 Tunneling Protocol, version 3 (L2TPv3) as the tunneling method.<br>• **mpls**—Specifies Multiprotocol Label Switching (MPLS) as the tunneling method.<br>• **manual**—Specifies that no signaling is to be used in the attachment circuit. This keyword places the router in xconnect configuration mode for manual configuration of the attachment circuit. Use this keyword to manually configure an AToM or L2TPv3 static pseudowire. |
| **pw-class** *pw-class-name* | (Optional) Specifies the pseudowire class for advanced configuration. |
| **sequencing** | (Optional) Sets the sequencing method to be used for packets received or sent. This keyword is not supported with the AToM Static Pseudowire Provisioning feature. |
| **transmit** | Sequences data packets received from the attachment circuit. |
| **receive** | Sequences data packets sent into the attachment circuit. |
| **both** | Sequences data packets that are both sent and received from the attachment circuit. |
| *pw-type* | (Optional) Pseudowire type. You can specify one of the following types:<br>• 4—Specifies Ethernet VLAN.<br>• 5—Specifies Ethernet port. |

## Command Default

The attachment circuit is not bound to the pseudowire.

## Command Modes

Connect configuration
Interface configuration (config-if)
l2transport configuration (for ATM)

## Command History

| Release | Modification |
|---|---|
| 12.0(23)S | This command was introduced. |
| 12.0(28)S | Support was added for Multilink Frame Relay connections. |
| 12.3(2)T | This command was integrated into Cisco IOS Release 12.3(2)T. |
| 12.2(25)S | This command was integrated into Cisco IOS Release 12.2(25)S. |
| 12.2(27)SBC | This command was integrated into Cisco IOS Release 12.2(27)SBC. |
| 12.4(11)T | This command was integrated into Cisco IOS Release 12.4(11)T. |
| 12.2(33)SRB | This command was updated to add support for AToM static pseudowires, and so that the remote router ID need not be the Label Distribution Protocol (LDP) router ID of the peer. |
| 12.2(33)SCC | This command was integrated into Cisco IOS Release 12.2(33)SCC. |
| 15.1(2)S | This command was updated to allow IPv6 address configurations in the ethernet sub-interface when the **xconnect** command is configured under a service instance on the main interface. This change only applies to platforms that support the **service instance** command on ethernet interfaces. |

## Usage Guidelines

The use of the **xconnect** command and the interface configuration mode bridge-group commands is not supported on the same physical interface.

The combination of the *peer-ip-address* and *vcid* arguments must be unique on the router. Each xconnect configuration must have a unique combination of *peer-ip-address* and *vcid* configuration.

**Note** If the remote router is a Cisco 12000 series Internet router, the *peer-ip-address* argument must specify a loopback address on that router.

The same *vcid* value that identifies the attachment circuit must be configured using the **xconnect** command on the local and remote PE router. The VC ID creates the binding between a pseudowire and an attachment circuit.

With the introduction of VPLS Autodiscovery in Cisco IOS Release 12.2(33)SRB, the remote router ID need not be the LDP router ID. The address you specify can be any IP address on the peer, as long as it is reachable. When VPLS Autodiscovery discovers peer routers for the VPLS, the peer router addresses might be any routable address.

**Note** The VPLS Autodiscovery feature is not supported with L2TPv3.

For L2TPv3, to manually configure the settings used in the attachment circuit, use the **manual** keyword in the **xconnect** command. This configuration is called a static session. The router is placed in xconnect configuration mode, and you can then configure the following options:

- Local and remote session identifiers (using the **l2tp id** command) for local and remote PE routers at each end of the session.

- Size of the cookie field used in the L2TPv3 headers of incoming (sent) packets from the remote PE peer router (using the **l2tp cookie local** command).
- Size of the cookie field used in the L2TPv3 headers of outgoing (received) L2TP data packets (using the **l2tp cookie remote** command).
- Interval used between sending hello keepalive messages (using the **l2tp hello** command).

For L2TPv3, if you do not enter the **encapsulation l2tpv3 manual** keywords in the **xconnect** command, the data encapsulation type for the L2TPv3 session is taken from the encapsulation type configured for the pseudowire class specified with the **pseudowire-class** *pw-class-name* command.

The **pw-class** keyword with the *pw-class-name* value binds the xconnect configuration of an attachment circuit to a specific pseudowire class. In this way, the pseudowire class configuration serves as a template that contains settings used by all attachment circuits bound to it with the **xconnect** command.

Software prior to Cisco IOS Release 12.2(33)SRB configured pseudowires dynamically using Label Distribution Protocol (LDP) or another directed control protocol to exchange the various parameters required for these connections. In environments that do not or cannot use directed control protocols, the **xconnect** command allows provisioning an AToM static pseudowire. Use the **manual** keyword in the **xconnect** command to place the router in xconnect configuration mode. MPLS pseudowire labels are configured using the **mpls label** and (optionally) **mpls control-word** commands in xconnect configuration mode.

## Examples

The following example configures xconnect service for an Ethernet interface by binding the Ethernet circuit to the pseudowire named 123 with a remote peer 10.0.3.201. The configuration settings in the pseudowire class named vlan-xconnect are used.

```
Router(config)# interface Ethernet0/0.1
Router(config-if)# xconnect 10.0.3.201 123 pw-class vlan-xconnect
```

The following example enters xconnect configuration mode and manually configures L2TPv3 parameters for the attachment circuit:

```
Router(config)# interface Ethernet 0/0
Router(config-if)# xconnect 10.0.3.201 123 encapsulation l2tpv3 manual pw-class ether-pw
Router(config-if-xconn) l2tp id 222 111
Router(config-if-xconn) l2tp cookie local 4 54321
Router(config-if-xconn) l2tp cookie remote 4 12345
Router(config-if-xconn) l2tp hello l2tp-defaults
```

The following example enters xconnect configuration mode and manually configures an AToM static pseudowire. The example shows the configuration for only one side of the connection; the configurations on each side of the connection must be symmetrical.

```
Router# configure terminal
Router(config)# interface Ethernet1/0
Router(config-if)# no ip address
Router(config-if)# xconnect 10.131.191.252 100 encapsulation mpls manual pw-class mpls
Router(config-if-xconn)# mpls label 100 150
Router(config-if-xconn)# exit
Router(config-if)# exit
```

The following example shows how to bind an attachment circuit to a pseudowire and configure an AToM service on a Cisco uBR10012 router:

```
Router# configure terminal
Router(config)# cable l2vpn 0000.396e.6a68 customer1
Router(config-l2vpn)# service instance 2000 Ethernet
Router(config-ethsrv)# xconnect 101.1.0.2 221 encapsulation mpls pw-type 4
```

## Related Commands

| Command | Description |
|---|---|
| **l2tp cookie local** | Configures the size of the cookie field used in the L2TPv3 headers of incoming packets received from the remote PE peer router. |
| **l2tp cookie remote** | Configures the size of the cookie field used in the L2TPv3 headers of outgoing packets sent from the local PE peer router. |
| **l2tp hello** | Specifies the use of a hello keepalive setting contained in a specified L2TP class configuration for a static L2TPv3 session. |
| **l2tp id** | Configures the identifiers used by the local and remote provider edge routers at each end of an L2TPv3 session. |
| **l2tp-class** | Configures a template of L2TP control plane configuration settings that can be inherited by different pseudowire classes. |
| **mpls control-word** | Enables the MPLS control word in an AToM static pseudowire connection. |
| **mpls label** | Configures an AToM static pseudowire connection by defining local and remote pseudowire labels. |
| **mpls label range** | Configures the range of local labels available for use on packet interfaces. |
| **pseudowire-class** | Configures a template of pseudowire configuration settings used by the attachment circuits transported over a pseudowire. |
| **show xconnect** | Displays information about xconnect attachment circuits and pseudowires. |

# xconnect logging pseudowire status

To enable system logging (syslog) reporting of pseudowire status events, use the **xconnect logging pseudowire status** command in global configuration mode. To disable syslog reporting of pseudowire status events, use the **no** form of this command.

**xconnect logging pseudowire status**

**no xconnect logging pseudowire status**

## Syntax Description

This command has no arguments or keywords.

## Defaults

Syslog reporting of pseudowire status events is off.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 12.0(31)S | This command was introduced. |
| 12.2(27)SBC | This command was integrated into Cisco IOS Release 12.2(27)SBC. |
| Cisco IOS XE Release 2.4 | This command was integrated into Cisco IOS XE Release 2.4. |
| 15.0(1)M | This command was integrated into a release earlier than Cisco IOS Release 15.0(1)M. |

## Examples

The following example enables syslog reporting of pseudowire status events:

```
Router# configure terminal
Router(config)# xconnect logging pseudowire status
```

## Related Commands

| Command | Description |
|---|---|
| **xconnect** | Binds an Ethernet, 802.1q VLAN, or Frame Relay attachment circuit to an L2TPv3 pseudowire for xconnect service and enters xconnect configuration mode. |