

Implementing MPLS OAM

- IP-Less MPLS-TP Ping and MPLS-TP Traceroute , on page 1
- MPLS LSP Ping, on page 1
- MPLS LSP Traceroute, on page 3
- MPLS OAM using nil FEC, on page 6
- Self-ping probe in MPLS-TE, on page 9

IP-Less MPLS-TP Ping and MPLS-TP Traceroute

In Label Switched Path (LSP) ping or traceroute with IP encapsulation over ACH, IP encapsulated ping or traceroute packets are sent over the MPLS LSP using the control channel (ACH). The application-level control channel in this case is the reverse path of the LSP using ACH. The on-demand ping or traceroute echo response message is sent on the reverse path of the LSP. The response uses ACH and is IP encapsulated. The destination address in the IP header is set to that of the sender of the echo request message, and the source address in the IP header is set to a valid address of the replying node.

- the reply mode is 4
- the node does not have a return MPLS LSP path to the echo request source.

MPLS LSP Ping

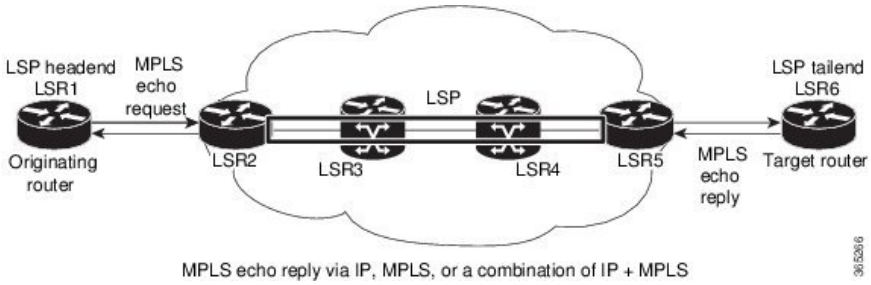
The MPLS LSP Ping feature is used to check the connectivity between Ingress LSR and egress LSRs along an LSP. MPLS LSP ping uses MPLS echo request and reply messages, similar to Internet Control Message Protocol (ICMP) echo request and reply messages, to validate an LSP. While ICMP echo request and reply messages validate IP networks, MPLS echo and reply messages validate MPLS networks. The MPLS echo request packet is sent to a target router through the use of the appropriate label stack associated with the LSP to be validated. Use of the label stack causes the packet to be forwarded over the LSP itself. The destination IP address of the MPLS echo request packet is different from the address used to select the label stack. The destination IP address is defined as a 127.x.y.z/8 address and it prevents the IP packet from being IP switched to its destination, if the LSP is broken.

An MPLS echo reply is sent in response to an MPLS echo request. The reply is sent as an IP packet and it is forwarded using IP, MPLS, or a combination of both types of switching. The source address of the MPLS echo reply packet is an address obtained from the router generating the echo reply. The destination address

is the source address of the router that originated the MPLS echo request packet. The MPLS echo reply destination port is set to the echo request source port.

The following figure shows MPLS LSP ping echo request and echo reply paths.

Figure 1: MPLS LSP Ping Echo Request and Reply Paths



By default, the **ping mpls ipv4** command tries to determine the Forwarding Equivalence Class (FEC) being used automatically. However, this is only applicable at head-end and works only if the FEC at the destination is same as the source. If the source and destination FEC types are not the same, the **ping mpls ipv4** command may fail to identify the targeted FEC type. You can overcome this limitation by specifying the FEC type in MPLS LSP ping using the **fec-type** command option. If the user is not sure about the FEC type at the transit or the destination, or it may change through network, use of the **generic** FEC type command option is recommended. Generic FEC is not coupled to a particular control plane and allows path verification when the advertising protocol is unknown, or may change during the path of the echo request. If you are aware of the destination FEC type, specify the target FEC as BGP or LDP.

Configuration Examples

This example shows how to use MPLS LSP ping to test the connectivity of an IPv4 LDP LSP. The destination is specified as a Label Distribution Protocol (LDP) IPv4 address.

```
RP/0/RP0/CPU0:router# ping mpls ipv4 10.1.1.2/32 verbose

Sun Nov 15 11:27:43.070 UTC

Sending 5, 100-byte MPLS Echos to 10.1.1.2/32,
      timeout is 2 seconds, send interval is 0 msec:

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, 'I' - unknown upstream index,
  'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.

!      size 100, reply addr 10.1.0.2, return code 3
!      size 100, reply addr 10.1.0.2, return code 3
!      size 100, reply addr 10.1.0.2, return code 3
!      size 100, reply addr 10.1.0.2, return code 3
!      size 100, reply addr 10.1.0.2, return code 3

Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/4 ms
```

In this example, the destination is specified as a Label Distribution Protocol (LDP) IPv4 prefix and Forwarding Equivalence Class (FEC) type is specified as generic.

```
RP/0/RP0/CPU0:router# ping mpls ipv4 10.1.1.2/32 fec-type generic

Wed Nov 25 03:36:33.143 UTC
Sending 5, 100-byte MPLS Echos to 10.1.1.2/32,
      timeout is 2 seconds, send interval is 0 msec:

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, 'I' - unknown upstream index,
  'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.

!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/3 ms
```

In this example, the destination is specified as a Label Distribution Protocol (LDP) IPv4 prefix and the FEC type is specified as BGP.

```
RP/0/RP0/CPU0:router# ping mpls ipv4 10.1.1.2/32 fec-type bgp

Wed Nov 25 03:38:33.143 UTC
Sending 5, 100-byte MPLS Echos to 10.1.1.2/32,
      timeout is 2 seconds, send interval is 0 msec:

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, 'I' - unknown upstream index,
  'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.

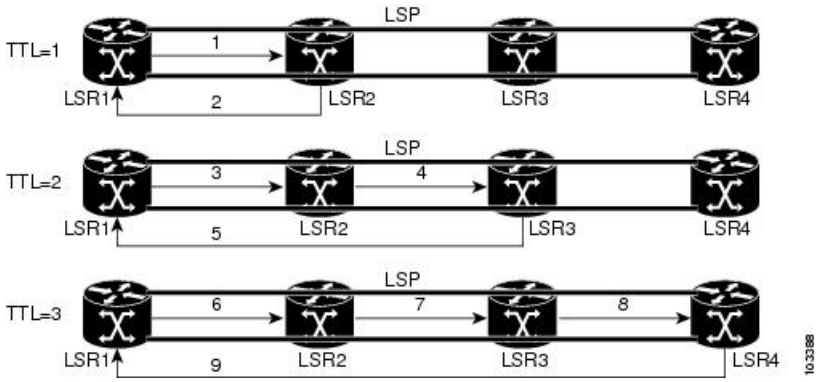
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/3 ms
```

MPLS LSP Traceroute

The MPLS LSP Traceroute feature is used to isolate the failure point of an LSP. It is used for hop-by-hop fault localization and path tracing. The MPLS LSP Traceroute feature relies on the expiration of the Time to Live (TTL) value of the packet that carries the echo request. When the MPLS echo request message hits a transit node, it checks the TTL value and if it is expired, the packet is passed to the control plane, else the message is forwarded. If the echo message is passed to the control plane, a reply message is generated based on the contents of the request message.

The following figure shows an MPLS LSP traceroute example with an LSP from LSR1 to LSR4.

Figure 2: MPLS LSP Traceroute



By default, the **traceroute mpls ipv4** command tries to determine the Forwarding Equivalence Class (FEC) being used automatically. However, this is only applicable at head-end and works only if the FEC at the destination is same as the source. If the source and destination FEC types are not the same, the **traceroute mpls ipv4** command may fail to identify the targeted FEC type. You can overcome this limitation by specifying the FEC type in MPLS LSP traceroute using the **fec-type** command option. If the user is not sure about the FEC type at the transit or the destination, or it may change through network, use of the **generic** FEC type command option is recommended. Generic FEC is not coupled to a particular control plane and allows path verification when the advertising protocol is unknown, or may change during the path of the echo request. If you are aware of the destination FEC type, specify the target FEC as BGP or LDP.

Configuration Examples

This example shows how to use the **traceroute** command to trace to a destination.

```
RP/0/RP0/CPU0:router# traceroute mpls ipv4 10.1.1.2/32 destination 127.0.0.3 127.0.0.6 2
Sat Jan 27 03:50:23.746 UTC

Tracing MPLS Label Switched Path to 10.1.1.2/32, 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, 'I' - unknown upstream index,
  'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.

Destination address 127.0.0.3
  0 10.2.1.2 MRU 1500 [Labels: 24000 Exp: 0]
L 1 10.2.1.1 MRU 1500 [Labels: implicit-null Exp: 0] 8 ms
! 2 10.1.0.2 3 ms

Destination address 127.0.0.5
  0 10.2.1.2 MRU 1500 [Labels: 24000 Exp: 0]
L 1 10.2.1.1 MRU 1500 [Labels: implicit-null Exp: 0] 5 ms
! 2 10.1.0.2 2 ms
```

This example shows how to use the **traceroute** command and how to specify the maximum number of hops for the traceroute to traverse by specifying the **ttl** value.

```
RP/0/RP0/CPU0:router# traceroute mpls ipv4 10.1.1.2/32 ttl 1
Sun Nov 15 12:20:14.145 UTC
Tracing MPLS Label Switched Path to 10.1.1.2/32, 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, 'I' - unknown upstream index,
  'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.

  0 10.1.0.1 MRU 1500 [Labels: implicit-null Exp: 0]
! 1 10.1.0.2 3 ms
```

This example shows how to use the **traceroute** command to trace to a destination and FEC type is specified as generic.

```
RP/0/RP0/CPU0:router# traceroute mpls ipv4 10.1.1.2/32 fec-type generic
Sun Nov 15 12:25:14.145 UTC
Tracing MPLS Label Switched Path to 10.1.1.2/32, 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, 'I' - unknown upstream index,
'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
0 10.12.12.1 MRU 1500 [Labels: implicit-null Exp: 0]
! 1 10.12.12.2 2 ms
```

This example shows how to use the **traceroute** command to trace to a destination and FEC type is specified as BGP.

```
RP/0/RP0/CPU0:router# traceroute mpls ipv4 10.1.1.2/32 fec-type bgp
Sun Nov 15 12:25:14.145 UTC
Tracing MPLS Label Switched Path to 10.1.1.2/32, 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, 'I' - unknown upstream index,
'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
0 10.12.12.1 MRU 1500 [Labels: implicit-null Exp: 0]
! 1 10.12.12.2 2 ms
```

MPLS OAM using nil FEC

Table 1: Feature History Table

Feature Name	Release Information	Feature Description
MPLS OAM using nil FEC	Release 26.1.1	Introduced in this release on: Centralized Systems (8400 [ASIC: K100])(select variants only*) *This feature is supported on Cisco 8404-SYS-D routers.
MPLS OAM using nil FEC	Release 25.4.1	Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*) *This feature is now supported on: • 8011-32Y8L2H2FH • 8011-12G12X4Y-A/D
MPLS OAM using nil FEC	Release 25.1.1	Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*) *This feature is supported on Cisco 8011-4G24Y4H-I routers.

Feature Name	Release Information	Feature Description
MPLS OAM using nil FEC	Release 24.4.1	Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100, K100])(select variants only*); Modular Systems (8800 [LC ASIC: Q100, P100])(select variants only*) The Nil-FEC LSP ping and traceroute operations are extensions of regular MPLS ping and traceroute. MPLS ping and traceroute requires at least one forwarding equivalence class (FEC) in the target FEC stack. This feature support is now extended to: • 8712-MOD-M • 8212-48FH-M • 8711-32FH-M • 88-LC1-52Y8H-EM • 88-LC1-12TH24FH-E • 88-LC1-36EH

The Nil-FEC LSP ping and traceroute operations are extensions of regular MPLS ping and traceroute. MPLS ping and traceroute requires at least one forwarding equivalence class (FEC) in the target FEC stack. In Nil-FEC ping and traceroute operations, an explicit FEC is not associated with the label. Nil-FEC LSP ping and traceroute support MPLS static LSPs and also act as an additional diagnostic tool for all other LSP types. Nil-FEC LSP ping and traceroute allow network operators to provide the ability to freely test any label stack by allowing them to specify the following:

- label stack
- outgoing interface
- nexthop address

The following table shows the syntax for the ping and traceroute commands.

Table 2: LSP Ping and Traceroute Nil FEC Commands

Command Syntax
ping mpls nil-fec labels {*label*[**,***label*]} [**output** {**interface** *tx-interface*} [**nexthop** *nexthop-ip-addr*]]
traceroute mpls nil-fec labels {*label*[**,***label*]} [**output** {**interface** *tx-interface*} [**nexthop** *nexthop-ip-addr*]]

Examples: LSP Ping Nil FEC and LSP Traceroute Nil FEC

The examples in this section use the following topology:

```
Node loopback IP address: 172.18.1.3   172.18.1.4   172.18.1.5   172.18.1.7
Node label:               16003        16004        16005        16007
Nodes:                    Arizona ---- Utah ------- Wyoming ---- Texas

Interface:            GigabitEthernet0/2/0/1   GigabitEthernet0/2/0/1
Interface IP address:         10.1.1.3              10.1.1.4
```

```
RP/0/RP0/CPU0:router-arizona# show mpls forwarding

Tue May  2 13:44:31.999 EDT
Local  Outgoing    Prefix             Outgoing     Next Hop        Bytes
Label  Label       or ID              Interface                    Switched
------ ----------- ------------------ ------------ --------------- ------------
16004  Pop         No ID              Gi0/2/0/1    10.1.1.4        1392
       Pop         No ID              Gi0/2/0/2    10.1.2.2        0
16005  16005       No ID              Gi0/2/0/0    10.1.1.4        0
       16005       No ID              Gi0/2/0/1    10.1.2.2        0
16007  16007       No ID              Gi0/2/0/0    10.1.1.4        4752
       16007       No ID              Gi0/2/0/1    10.1.2.2        0
```

This example shows how to use Nil-FEC LSP ping to test a label stack.

```
RP/0/RP0/CPU0:router-arizona# ping mpls nil-fec labels 16005,16007 output interface
GigabitEthernet 0/2/0/1 nexthop 10.1.1.4 repeat 1
Sending 1, 72-byte MPLS Echos with Nil FEC labels 16005,16007,
     timeout is 2 seconds, send interval is 0 msec:

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,
  'd' - see DDMAP for return code,
  'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.
!
Success rate is 100 percent (1/1), round-trip min/avg/max = 1/1/1 ms
 Total Time Elapsed 0 ms
```

This example shows how to use Nil-FEC LSP traceroute for a label stack.

```
RP/0/RP0/CPU0:router-arizona# traceroute mpls nil-fec labels 16005,16007 output interface
GigabitEthernet 0/2/0/1 nexthop 10.1.1.4
Tracing MPLS Label Switched Path with Nil FEC labels 16005,16007, 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,
  'd' - see DDMAP for return code,
```

```
  'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.
  0 10.1.1.3 MRU 1500 [Labels: 16005/16007/explicit-null Exp: 0/0/0]
L 1 10.1.1.4 MRU 1500 [Labels: implicit-null/16007/explicit-null Exp: 0/0/0] 1 ms
L 2 10.1.1.5 MRU 1500 [Labels: implicit-null/explicit-null Exp: 0/0] 1 ms
! 3 10.1.1.7 1 ms
```

Self-ping probe in MPLS-TE

A self-ping probe is a process that

- sends a test packet, usually UDP, from the ingress Label Edge Router (LER) across a newly established Label Switched Path (LSP)
- receives the returned packet from the egress LER, and
- verifies that the LSP is forwarding data correctly.

Table 3: Feature History Table

Feature Name	Release Information	Feature Description
Self-ping probe for reoptimized LSP	Release 25.4.1	Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*)(select variants only*) *This feature is supported on: • 8011-32Y8L2H2FH • 8011-12G12X4Y-A/D
Self-ping probe for current LSP	Release 25.3.1	Introduced in this release on: Fixed Systems (8200 [ASIC: Q100, Q200, P100], 8700 [ASIC: P100, K100], 8010 [ASIC: A100]); Centralized Systems (8600 [ASIC:Q200]); Modular Systems (8800 [LC ASIC: Q100, Q200, P100]) The self-ping probe functionality is extended to support the current LSP that allows the network to promptly confirm the readiness of existing LSP to handle traffic. This immediate verification helps prevent traffic drops by ensuring that the data path is fully operational before forwarding actual user traffic. The self-ping option is configured under a named tunnel and the operational status of the tunnel is determined by probe packet or during session timeout.

Self-ping probe for reoptimized LSP	Release 25.1.1	Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*) *This feature is supported on: • 8011-4G24Y4H-I • 8712-MOD-M
Self-ping probe for reoptimized LSP	Release 24.4.1	Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*) *This feature is supported on: • 8212-48FH-M • 8711-32FH-M • 88-LC1-36EH • 88-LC1-12TH24FH-E • 88-LC1-52Y8H-EM
Self-ping probe for reoptimized LSP	Release 7.5.3	In a reoptimized LSP, the reopt-install timer helps to prevent traffic issue. This process is made possible by enabling the label edge router (LER) to send self-ping probes over the reoptimized LSP to the ingress LER. As soon the probe reaches the LER, there's confirmation that the RSVP programming is complete along the path. Post this confirmation, the LER switches traffic to the reoptimized LSP with no drop in traffic. This feature introduces the **self-ping** keyword in the **named-tunnels tunnel-te** command.

Supported scenarios in self-ping probe

- Self-ping probe for current LSP—refers to the active or presently established Label Switched Path being used for forwarding traffic between a specific pair of ingress and egress routers in an MPLS network.
- Self-ping probe for reoptimized LSP—refers to a Label Switched Path that has been recalculated and reestablished to improve its performance, efficiency, or resource utilization.

Self-ping probe for current LSP

When a tunnel lacks an active path—such as when it is newly created or has just recovered from a down state—a new LSP is established. In the traditional process, the tunnel is considered operational and ready for use as soon as RSVP signaling is completed. However, there can be delays in programming the forwarding state at some nodes along the path, which means that if traffic is sent immediately, it may be lost if those nodes are not yet fully ready.

To address this, self-ping introduces an extra verification step to LSP setup.The ingress LER sends a test probe to the egress LER. When the probe is successfully returned to the ingress LER, the tunnel is ready for carrying traffic. This ensures that the tunnel is ready for data forwarding and prevents any traffic loss.

Self-ping probe for reoptimized LSP

During the reoptimization of an LSP, the LER must wait until the reoptimized path is fully established before switching traffic onto the new LSP.

If the LER does not wait long enough and RSVP programming for MPLS forwarding is not complete at a specific hop along the path, traffic may be dropped at that hop.

The LER operates at the edge of an MPLS network and serves as the entry and exit point for the network.

Reopt-install timer function

The edge router is equipped with an internal reopt-install timer that provides sufficient time for the reoptimized path to be established before directing traffic onto it. If the LER waits too long before switching, traffic continues to use the old path, potentially causing congestion.

To avoid potential traffic loss, operators often use a cautious approach by waiting longer before transitioning traffic to the reoptimized LSP. The self-ping can bypass long waiting time and switch traffic to reoptimitzed LSP as soon as the self-ping probe proves that the LSP is operational.

Restriction for self-ping probe support

- Self-ping is supported only with named tunnels.
- Self -ping is not applied for LSP which has **1-hop long** path even it is configured for the tunnel.

How self-ping probe works

Summary

The self-ping probe mechanism dynamically and reliably detects when a reoptimized LSP is ready for traffic switching. The key components involved in the process are:

- Ingress Label Edge Router (LER): Sends self-ping probes over the current or reoptimized LSP.
- Egress Label Edge Router (LER): Receives probes and sends responses.
- Self-ping probe: A probe sent once per second to verify LSP readiness.

The process involves these stages:

1. The ingress LER sends a self-ping probe to the egress LER over the current or reoptimized LSP at a fixed rate of one probe per second. This frequency is consistent regardless of the LSP type and is not configurable.
2. Upon receiving the probe response, the ingress LER declares the LSP as Up.
3. The ingress LER immediately switches traffic to the new, reoptimized path.
4. After a successful switchover, no further probes are sent for that LSP.

5. If the self-ping mechanism sends the specified number of probes without receiving a response, additional handling occurs.

Result:

This process ensures timely and reliable detection of reoptimized LSP readiness, enabling immediate traffic switchover to the optimal path and minimizing downtime.

Configure the self-ping probe

The purpose of this task is to enable self-ping probe in current and reoptimized LSPs.

Procedure

Step 1 Configure an MPLS-TE tunnel and enable the self-ping feature to verify the tunnel's path before traffic is sent over it.

Example:

```
Router# configure
Router(config)# mpls traffic-eng
Router(config-mpls-te)# named-tunnels tunnel-te ABC
Router(config-te-tun-name)# self-ping
```

Step 2 *Optional:* Set the maximum number of self-ping probes.

Example:

```
Router(config-te-tun-name)# max-count 10
Router(config-te-tun-name)# commit
```

The ingress LER sends up to 10 probes to check the LSP before making a decision.

If not set manually, the default number of probes sent is 60.

Step 3 Run the **show mpls traffic-eng tunnels name** *tunnel-name* detail to verify the readiness of the MPLS-TE tunnel.

Example:

```
Router# show mpls traffic-eng tunnels name ABC detail

Name: ABC  Destination: 192.168.0.4  Ifhandle:0x2d0
  Tunnel-ID: 32783
  Status:
    Admin:    up Oper:   up   Path:  valid   Signalling: connected

    G-PID: 0x0800 (derived from egress interface properties)
    Bandwidth Requested: 0 kbps  CT0
    Creation Time: Thu Apr  7 14:54:30 2022 (00:00:01 ago)
  Config Parameters:
    ...
Load-interval: 300 seconds
    Auto-bw: disabled
    Auto-Capacity: Disabled:
    Self-ping: Enabled
      Maximum-probes: 10
      Probes-period: 1 second(s)
    Fast Reroute: Disabled, Protection Desired: None
    Path Protection: Not Enabled
    BFD Fast Detection: Disabled
```

```
    Reoptimization after affinity failure: Enabled
    Soft Preemption: Disabled
  Self-ping:
    Status: Not executed
  SNMP Index: 0
  Binding SID: 0
  History:
    Tunnel has been up for: 00:00:00 (since Thu Jan 01 01:00:00 BST 1970)
    Current LSP:
      Uptime: 00:00:00 (since Thu Jan 01 01:00:00 BST 1970)
  Current LSP Info:
    Instance: 2, Signaling Area: OSPF 100 area 0
    Uptime: 00:00:00 (since Thu Apr 07 14:54:31 BST 2022)
    Outgoing Interface: GigabitEthernet0/2/0/0, Outgoing Label: 24000
    Next Hop: 10.10.10.2      Neighbor Next Hop: 0.0.0.0
    Router-IDs: local      192.168.0.1
                downstream 192.168.0.2
...
Displayed 1 (of 1) heads, 0 (of 0) midpoints, 0 (of 0) tails
Displayed 1 up, 0 down, 0 recovering, 0 recovered heads
```

Step 4 Run the **show mpls traffic-eng self-ping statistics** command to provide an overview of the activity, success, and failure rates of MPLS-TE tunnel self-ping operations.

Example:

```
Router# show mpls traffic-eng self-ping statistics
Wed Jun 15 14:24:29.316 BST
Self-Ping Statistics:
  Collected since: Tue Jun 14 09:35:52 2022 (1d04h ago)
  Operations:
    Started 2
    Running 0
    Successful 1
    Timed-out 1
    Terminated 0
  Probes sent 11
  Probes failed 0
  Received responses 1 (Average response time 00:00:00)
  Mismatched responses 0
```

You can clear the statistics using **clear mpls traffic-eng self-ping statistics** command. If there are any active sessions running, this field is not reset to zero, and the started count is also set to this value.

Self-ping probe status

During the self-ping procedure, the tunnel goes through various stages and the LER shows the status when you run the **show mpls traffic-eng tunnels detail** command.

This table describes the status and outcomes of self-ping probes in the MPLS RSVP-TE environment.

Table 4: Self-ping probe status

Self-ping status	Description	Sample ouput
Not executed	Although self-ping is configured on the ingress LER, the request to setup new LSP is not triggered.	`Self-ping:` `  Status: Not executed`

Self-ping status	Description	Sample ouput

Underway	LSP verification is ongoing and the self-ping is actively sending probes and waiting to receive the response.	`Self-ping:`
		`  Status: Underway`
		`  LSP-ID: 6`
		`  Started: 00:00:00 (since Thu Apr 07 15:03:33 BST 2022)`

Succeeded	The ingress LER receives the response from the egress LER and and LSP is ready to carry the traffic.	`Self-ping:`
		`  Status: Succeeded (in 0 seconds)`
		`  LSP-ID: 4`
		`  Probes sent: 1`
		`  Started: 00:00:00 (since Thu Apr 07 15:02:23 BST 2022)`
		`  Stopped: 00:00:00 (since Thu Apr 07 15:02:23 BST 2022)`
		`  Response Received: 00:00:00 (since Thu Apr 07 15:02:23`
		`  BST 2022)`

Timed-out	After reaching the maximum number of probes with no response, the ingress LER falls back to the reopt-install timer like behavior and LSP is considered active, and traffic is switched to the reoptimized LSP after the reopt-install timer expires.	`Self-ping:`
		`  Status: Timed-out (in 9 seconds)`
		`  LSP-ID: 6`
		`  Probes sent: 10`
		`  Started: 00:00:43 (since Thu Apr 07 15:03:33 BST 2022)`
		`  Stopped: 00:00:34 (since Thu Apr 07 15:03:42 BST 2022)`

Self-ping status	Description	Sample ouput

Terminated	The LER terminates self-ping execution when:
	• LSP under self-ping is obsolete. For example:
	  • The LSP under setup has failed.
	  • LSP path becomes invalid.
	  • New LSP is obsolete.
	  • Tunnel is being shut down.
	  • Self-ping configuration is removed or whole tunnel is removed.
	• The self-ping process is managed by the self-ping watchdog timer. If self-ping does not return a success or timeout result, the watchdog timer will forcefully stop and clean up the self-ping session, and the tunnel will proceed as if a self-ping timeout had occurred.
		`Self-ping:`
		`  Status: Terminated (in 20 seconds)`
		`  LSP-ID: 8`
		`  Started: 00:00:21 (since Thu Apr 07 15:05:44 BST 2022)`
		`  Stopped: 00:00:01 (since Thu Apr 07 15:06:04 BST 2022)`