



## Segment Routing MPLS OAM Support

Segment Routing Operations, Administration, and Maintenance (OAM) helps service providers to monitor label-switched paths (LSPs) and quickly isolate forwarding problems to assist with fault detection and troubleshooting in the network. The Segment Routing OAM feature provides support for Nil-FEC (forwarding equivalence classes) LSP Ping and Traceroute, IGP prefix SID FEC type, and partially IGP adjacency-SID FEC type for SR-TE functionality.

- [Restrictions for Segment Routing OAM MPLS Support, on page 1](#)
- [Information About Segment Routing MPLS OAM Support, on page 2](#)
- [How to Diagnose Segment Routing with LSP Ping and Trace Route Nil FEC Target, on page 3](#)
- [Example for LSP Ping Nil FEC Target Support, on page 4](#)
- [Path Validation in Segment Routing Network, on page 6](#)
- [Configuring Segment Routing MPLS Traffic Engineering for MPLS Ping and Traceroute, on page 8](#)
- [Configuring Segment Routing MPLS IGP for MPLS Ping and Traceroute, on page 8](#)
- [Additional References for Segment Routing OAM Support, on page 9](#)
- [Feature Information for Segment Routing OAM Support, on page 9](#)

## Restrictions for Segment Routing OAM MPLS Support

- Ping and traceroute are unsupported with SR-TE static auto tunnel, BGP Dynamic TE, and on-demand next hop auto tunnels.
- Strict-SID option is not supported by the path installed by OSPF.
- MPLS traceroute does not support popping of two explicit null labels in one node.
- Rerouting the path to IP over MPLS segment without using Layer3 VPN is not supported due to IP routing destination not being a MPLS FEC.

# Information About Segment Routing MPLS OAM Support

## Segment Routing OAM Support

The Nil-FEC LSP ping and traceroute operations are extensions of regular MPLS ping and traceroute. Nil-FEC LSP Ping/Trace functionality support Segment Routing and MPLS Static. It also act as an additional diagnostic tool for all other LSP types. This feature allows operators to test any label stack to specify the following:

- label stack
- outgoing interface
- nexthop address

In the case of segment routing, each segment nodal label and adjacent label along the routing path is put into the label stack of an echo request message from initiator Label Switch Router (LSR); MPLS data plane forward this packet to the label stack target, and the label stack target reply the echo message back.

## Benefits of Segment Routing OAM Support

- The feature enables the MPLS OAM functionality in the Segment Routing Network where the traffic is engineering via SR-TE tunnels or native SR forwarding.
- In traditional MPLS networks, source node chooses the path based on hop by hop signaling protocols such as LDP or RSVP-TE. In Segment Routing Networks, the path is specified by set of segments which are advertised by the IGP protocols (currently OSPF and ISIS).
- As the volume of services offered using SR increase, it is important that the operator essentially is able to do the connectivity verification and the fault isolation in the SR architecture.
- The segment assignment is not based on hop by hop protocols as in traditional MPLS network, any broken transit node could lead in traffic blackholing, which could lead to undesired behavior.
- Both SR and SR-TE supports load balancing, it is important to trace all the ECMP paths available between source and target routers. The features offers the multipath traceroute support for both TE and native SR paths.
- The following are the main benefits of Segment Routing-OAM Support:
  - **Operations:** Network monitoring and fault management.
  - **Administration:** Network discovery and planning.
  - **Maintenance:** Corrective and preventive activities, minimize occurrences and impact of failures.

## Segment Routing MPLS Ping

MPLS ping and traceroute are extendable by design. You can add SR support by defining new FECs and/or additional verification procedures. MPLS ping verifies MPLS data path and performs the following:

- Encapsulates echo request packet in MPLS labels.

- Measures coarse round trip time.
- Measures coarse round trip delay.

## Segment Routing MPLS Traceroute

MPLS ping and traceroute are extendable by design. You can add SR support by defining new forwarding equivalence classes (FECs) and/or additional verification procedures. MPLS traceroute verifies forwarding and control plane at each hop of the LSP to isolate faults. Traceroute sends MPLS echo requests with monotonically increasing time-to-live (TTL), starting with TTL of 1. Upon TTL expiry, transit node processes the request in software and verifies if it has an LSP to the target FEC and intended transit node. The transit node sends echo reply containing return code specifying the result of above verification and label stack to reach the next-hop, as well as ID of the next-hop towards destination, if verification is successful. Originator processes echo reply to build the next echo request containing TTL+1. Process is repeated until the destination replies that it is the egress for the FEC.

## LSP Ping Operation for Nil FEC target

The LSP Ping/Traceroute is used in identifying LSP breakages. The nil-fec target type can be used to test the connectivity for a known label stack. Follow the existing LSP ping procedure (for more information, refer [MPLS LSP Ping/Traceroute](#)), with the following modifications:

- Build the echo request packet with the given label stack.
- Append explicit null label at the bottom of the label stack.
- Build echo request FTS TLV with target FEC Nil FEC and label value set to the bottom label of the label stack, which is explicit-null.

# How to Diagnose Segment Routing with LSP Ping and Trace Route Nil FEC Target

## Using LSP Ping for Nil FEC Target

The Nil FEC LSP ping and traceroute operation are simply extension of regular MPLS ping and trace route. **nil-fec labels <label, label...>** is added to the ping mpls command. This command sends an echo request message with MPLS label stack as specified and add another explicit null at bottom of the stack.

```
ping mpls nil-fec labels <comma separated labels> output interface <tx-interface> nexthop
<nexthop ip addr>
[repeat <count>]
[size <size> | sweep <min_size> <max_size> <increment>]
[timeout <seconds>]
[interval <milliseconds>]
[destination <addr_start> [<addr_end> [<addr_incr_mask> | <addr_incr>]]]
[source <addr>]
[exp <exp-value>]
[pad <pattern>]
[ttl <t1>]
```

```
[reply [mode [ipv4 | router-alert | no-reply]]
[dscp <dscp-bits>]
[pad-tlv]]
[verbose]
[force-disposition ra-label]
{dsmap | ddmmap [l2ecmp]} [hashkey {none | {ipv4 | ipv4-label-set {bitmap <bitmap_size>}}}]
```

For more information, refer [ping mpls](#).

## Using LSP Traceroute for Nil FEC Target

```
trace mpls nil-fec labels <comma separated labels> output interface <tx-interface> nexthop
  <nexthop ip addr>
[timeout <seconds>]
[destination <addr_start> [<addr_end> [<addr_incr_mask> | <addr_incr>]]]
[source <addr> ]
[exp <exp-value>]
[ttl <ttl-max>]
[reply [mode [ipv4 | router-alert | no-reply]]
[dscp <dscp-bits>]
[pad-tlv]]
[flags {fec | ttl}]
[hashkey ipv4 | ipv4-label-set {bitmap <bitmap_size>}]
```

For more information, refer to the [traceroute mpls](#).

## Example for LSP Ping Nil FEC Target Support

```
Node loopback IP address: 1.1.1.3          1.1.1.4          1.1.1.5
                        1.1.1.7
Node label:                16004          16005
                        16007
Nodes:                      Arizona ----- Utah ----- Wyoming
----- Texas
Interface:                   Eth1/0          Eth1/0
Interface IP address:        30.1.1.3      30.1.1.4
```

```
Device#sh mpls forwarding-table
Local   Outgoing  Prefix          Bytes Label  Outgoing  Next Hop
Label   Label    or Tunnel Id   Switched     interface
16      Pop Label 3333.3333.0000-Et1/0-30.1.1.3  \
        0
        Et1/0    30.1.1.3
17      Pop Label 5555.5555.5555-Et1/1-90.1.1.5  \
        0
        Et1/1    90.1.1.5
18      Pop Label 3333.3333.0253-Et0/2-102.102.102.2 \
        0
        Et0/2    102.102.102.2
19      Pop Label 9.9.9.4/32     0
        Et0/2    102.102.102.2
20      Pop Label 1.1.1.5/32     0
        Et1/1    90.1.1.5
21      Pop Label 1.1.1.3/32     0
        Et1/0    30.1.1.3
22      Pop Label 16.16.16.16/32 0
        Et1/0    30.1.1.3
23      Pop Label 16.16.16.17/32 0
        Et1/0    30.1.1.3
24      Pop Label 17.17.17.17/32 0
        Et1/0    30.1.1.3
25      20        9.9.9.3/32     0
        Et1/0    30.1.1.3
26      21        1.1.1.6/32     0
        Et1/0    30.1.1.3
27      24        1.1.1.2/32     0
        Et1/0    30.1.1.3
        28        1.1.1.2/32     0
        Et1/1    90.1.1.5
28      18        1.1.1.7/32     0
        Et1/1    90.1.1.5
29      27        9.9.9.7/32     0
        Et1/1    90.1.1.5
30      Pop Label 55.1.1.0/24    0
        Et1/1    90.1.1.5
```

Local Label	Outgoing Label	Prefix or Tunnel Id	Bytes Switched	Label	Outgoing interface	Next Hop
31	Pop Label	19.1.1.0/24	0		Et1/0	30.1.1.3
32	Pop Label	100.1.1.0/24	0		Et1/0	30.1.1.3
33	Pop Label	100.100.100.0/24	0		Et1/0	30.1.1.3
34	Pop Label	110.1.1.0/24	0		Et1/0	30.1.1.3
35	28	10.1.1.0/24	0		Et1/0	30.1.1.3
36	29	101.101.101.0/24	0		Et1/0	30.1.1.3
37	29	65.1.1.0/24	0		Et1/1	90.1.1.5
38	33	104.104.104.0/24	0		Et1/0	30.1.1.3
	39	104.104.104.0/24	0		Et1/1	90.1.1.5
39	30	103.103.103.0/24	0		Et1/1	90.1.1.5
16005	Pop Label	1.1.1.5/32	1782		Et1/1	90.1.1.5
16006	16006	1.1.1.6/32	0		Et1/0	30.1.1.3
16007	16007	1.1.1.7/32	0		Et1/1	90.1.1.5
16017	16017	17.17.17.17/32	0		Et1/0	30.1.1.3
16250	16250	9.9.9.3/32	0		Et1/0	30.1.1.3
16252	16252	9.9.9.7/32	0		Et1/1	90.1.1.5
16253	Pop Label	9.9.9.4/32	0		Et0/2	102.102.102.2
17000	17000	16.16.16.16/32	0		Et1/0	30.1.1.3
17002	17002	1.1.1.2/32	0		Et1/0	30.1.1.3
	17002	1.1.1.2/32	0		Et1/1	90.1.1.5

```
Device#ping mpls nil-fec labels 16005,16007 output interface ethernet 1/0 nexthop 30.1.1.4
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,
'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.
```

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

```
Device#traceroute mpls nil-fec labels 16005,16007 output interface ethernet 1/0 nexthop
30.1.1.4
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,
'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 30.1.1.3 MRU 1500 [Labels: 16005/16007/explicit-null Exp: 0/0/0]
L 1 30.1.1.4 MRU 1500 [Labels: implicit-null/16007/explicit-null Exp: 0/0/0] 1 ms
L 2 90.1.1.5 MRU 1500 [Labels: implicit-null/explicit-null Exp: 0/0] 1 ms
! 3 55.1.1.7 1 ms
```

## Path Validation in Segment Routing Network

The MPLS OAM mechanisms help with fault detection and isolation for a MPLS data-plane path by the use of various target FEC stack sub-TLVs that are carried in MPLS echo request packets and used by the responder for FEC validation. While it is obvious that new sub-TLVs need to be assigned for segment routing, the unique nature of the segment routing architecture raises the need for additional operational considerations for path validation.

The forwarding semantic of Adjacency Segment ID is to pop the Segment ID and send the packet to a specific neighbor over a specific link. A malfunctioning node may forward packets using Adjacency Segment ID to an incorrect neighbor or over an incorrect link. The exposed Segment ID (of an incorrectly forwarded Adjacency Segment ID) might still allow such packet to reach the intended destination, although the intended strict traversal has been broken. MPLS traceroute may help with detecting such a deviation.

The format of the following Segment ID sub-TLVs follows the philosophy of Target FEC Stack TLV carrying FECs corresponding to each label in the label stack. This allows LSP ping/traceroute operations to function when Target FEC Stack TLV contains more FECs than received label stack at responder nodes. Three new sub-TLVs are defined for Target FEC Stack TLVs (Type 1), Reverse-Path Target FEC Stack TLV (Type 16) and Reply Path TLV (Type 21).

sub-Type	Value Field
34	IPv4 IGP-Prefix Segment ID
35	IPv6 IGP-Prefix Segment ID
36	IGP-Adjacency Segment ID

## MPLS Ping and Traceroute for IGP Prefix-SID FEC Type

MPLS ping and traceroute operations for prefix SID are supported for various IGP scenarios, for example:

- Within an IS-IS level or OSPF area
- Across IS-IS levels or OSPF areas
- Route redistribution from IS-IS to OSPF and from OSPF to IS-IS

The MPLS LSP Ping feature is used to check the connectivity between ingress Label Switch Routers (LSRs) 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. The destination IP address of the MPLS echo request packet is different from the address used to select the label stack.

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.

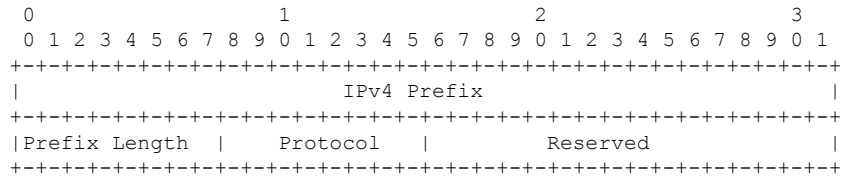
The MPLS LSP Tree Trace (traceroute multipath) operation is also supported for IGP Prefix SID. MPLS LSP Tree Trace provides the means to discover all possible equal-cost multipath (ECMP) routing paths of an LSP to reach a destination Prefix SID. It uses multipath data encoded in echo request packets to query for the load-balancing information that may allow the originator to exercise each ECMP. When the packet TTL expires at the responding node, the node returns the list of downstream paths, as well as the multipath

information that can lead the operator to exercise each path in the MPLS echo reply. This operation is performed repeatedly for each hop of each path with increasing TTL values until all ECMP are discovered and validated.

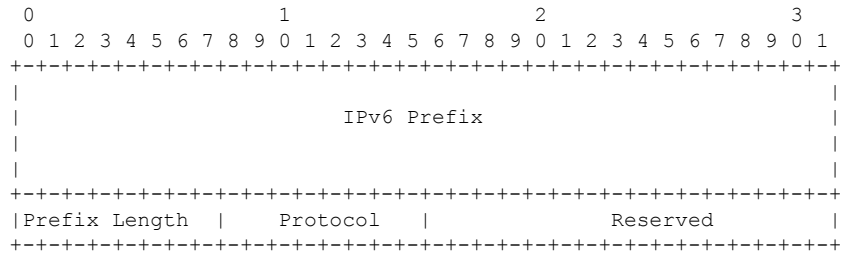
MPLS echo request packets carry Target FEC Stack sub-TLVs. The Target FEC sub-TLVs are used by the responder for FEC validation. The IGPIPv4 prefix sub-TLV has been added to the Target FEC Stack sub-TLV. The IGP IPv4 prefix sub-TLV contains the prefix SID, the prefix length, and the protocol (IS-IS or OSPF).

The network node which advertised the Node Segment ID is responsible for generating a FEC Stack Change sub-TLV with pop operation type for Node Segment ID, regardless of whether penultimate hop popping (PHP) is enabled or not.

The format is as below for IPv4 IGP-Prefix Segment ID:



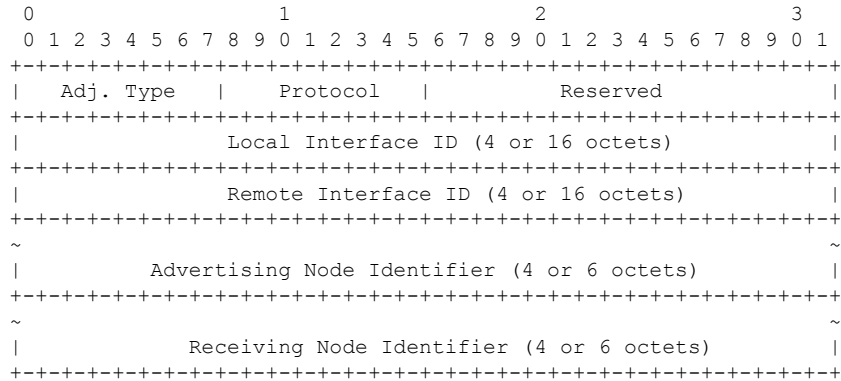
The format is as below for IPv6 IGP-Prefix Segment ID:



## MPLS Ping and Traceroute for IGP-Adjacency Segment ID

The network node that is immediate downstream of the node which advertised the Adjacency Segment ID is responsible for generating FEC Stack Change sub-TLV for "POP" operation for Adjacency Segment ID.

The format is as below for IGP-adjacency SID:



## Configuring Segment Routing MPLS Traffic Engineering for MPLS Ping and Traceroute

```

ping mpls traffic-eng tunnel <tun-id>
[repeat <count>]
[size <size> | sweep <min_size> <max_size> <increment>]
[timeout <seconds>]
[interval <milliseconds>]
[destination <addr_start> [<addr_end> [<addr_incr_mask> | <addr_incr>]]]
[source <addr>]
[exp <exp-value>]
[pad <pattern>]
[ttl <ttl>]
[reply [mode [ipv4 | router-alert | no-reply]]]
[dscp <dscp-bits>]
[pad-tlv]
[verbose]
[output {interface <tx-interface>} [nexthop <nexthop ip addr>]]
[{dsmap | ddmap [l2ecmp]} [hashkey {none | {ipv4 | ipv4-label-set {bitmap <bitmap_size>}}}]

traceroute mpls [multipath] traffic-eng <tunnel-interface>
[timeout <seconds>]
[destination <addr_start> [<addr_end> [<addr_incr_mask> | <addr_incr>]]]
[source <addr> ]
[exp <exp-value>]
[ttl <ttl-max>]
[reply [mode [ipv4 | router-alert | no-reply]]]
[pad-tlv]
[output {interface <tx-interface>} [nexthop <nexthop ip addr>]]
[flags {fec | ttl}]
[hashkey ipv4 | ipv4-label-set {bitmap <bitmap_size>}]

```

## Configuring Segment Routing MPLS IGP for MPLS Ping and Traceroute

```

ping mpls ipv4 <prefix/prefix_length> [fec-type [ldp | bgp | generic | isis | ospf]]
[sr-path-type [ip | sid | strict-sid]]
[destination <addr_start> [<addr_end> [<addr_incr_mask> | <addr_incr>]]]
[source <addr>]
[exp <exp-value>]
[pad <pattern>]
[ttl <ttl>]
[reply [mode [ipv4 | router-alert | no-reply]]]
[dscp <dscp-bits>]
[pad-tlv]
[verbose]
[output {interface <tx-interface>} [nexthop <nexthop ip addr>]]
[{dsmap | ddmap [l2ecmp]} [hashkey {none | {ipv4 | ipv4-label-set {bitmap <bitmap_size>}}}]

traceroute mpls [multipath] ipv4 <prefix/prefix_length> [fec-type [ldp | bgp | generic |
isis | ospf]] [sr-path-type [ip | sid | strict-sid]]
[timeout <seconds>]
[destination <addr_start> [<addr_end> [<addr_incr_mask> | <addr_incr>]]]
[source <addr> ]

```



```
[exp <exp-value>]
[ttl <ttl-max>]
[reply [mode [ipv4 | router-alert | no-reply]]]
[pad-tlv]]
[output {interface <tx-interface>} [nexthop <nexthop ip addr>]]
[flags {fec | ttl}]
[hashkey ipv4 | ipv4-label-set {bitmap <bitmap_size>}]]
```

- fec-type: IPv4 Target FEC type, use head end auto detected FEC type by default.
- sr-path-type: Segment routing path type selection algorithm. Use IP imposition path, when option is specified.

## Additional References for Segment Routing OAM Support

### Related Documents

Related Topic	Document Title
Cisco IOS commands	<a href="#">Cisco IOS Master Commands List, All Releases</a>

## Feature Information for Segment Routing OAM Support

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

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

**Table 1: Feature Information for Segment Routing OAM Support**

Feature Name	Releases	Feature Information
Segment Routing OAM Support	Cisco IOS XE Release 3.17 S	The Segment Routing OAM feature provides support for Nil-FEC (forwarding equivalence classes) LSP Ping and Traceroute functionality.  The Nil-FEC LSP ping and traceroute operation are simply extension of regular MPLS ping and trace route.

