



MPLS EM—MPLS LSP Multipath Tree Trace

First Published: December 4, 2006

Last Updated: February 27, 2009

The MPLS EM—MPLS LSP Multipath Tree Trace feature provides the means to discover all possible equal-cost multipath (ECMP) routing paths of a label switched path (LSP) between an egress and ingress router. Once discovered, these paths can be retested on a periodic basis using Multiprotocol Label Switching (MPLS) LSP ping or traceroute. This feature is an extension to the MPLS LSP traceroute functionality for the tracing of IPv4 LSPs.

You can use the MPLS EM—MPLS LSP Multipath Tree Trace feature to discover all paths for an IPv4 LSP.

This implementation of the MPLS EM—MPLS LSP Multipath Tree Trace feature is based on RFC 4379, *Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures*.

For information on the use of MPLS LSP ping and traceroute, see the [MPLS Embedded Management—LSP Ping/Traceroute and AToM VCCV](#) feature module.

Cisco IOS MPLS Embedded Management (EM) is a set of standards and value-added services that facilitate the deployment, operation, administration, and management of MPLS-based networks according to the fault, configuration, accounting, performance, and security (FCAPS) model.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the [“Feature Information for MPLS EM—MPLS LSP Multipath Tree Trace”](#) section on page 33.

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



Americas Headquarters:

Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Contents

- Prerequisites for MPLS EM—MPLS LSP Multipath Tree Trace, page 2
- Restrictions for MPLS EM—MPLS LSP Multipath Tree Trace, page 2
- Information About MPLS EM—MPLS LSP Multipath Tree Trace, page 3
- How to Configure MPLS EM—MPLS LSP Multipath Tree Trace, page 4
- Configuration Examples for MPLS EM—MPLS LSP Multipath Tree Trace, page 22
- Additional References, page 31
- Command Reference, page 32
- Feature Information for MPLS EM—MPLS LSP Multipath Tree Trace, page 33
- Glossary, page 35

Prerequisites for MPLS EM—MPLS LSP Multipath Tree Trace

The following are prerequisites for using the MPLS EM—MPLS LSP Multipath Tree Trace feature:

- You must understand the concepts and know how to use MPLS LSP ping or traceroute as described in the *MPLS LSP Ping/Traceroute for LDP/TE, and LSP Ping for VCCV* document.
- The routers in your network must be using an implementation based on RFC 4379, *Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures*.
- You should know the following about your MPLS network:
 - The topology
 - The number of links in your network
 - The expected number of LSPs, and how many LSPs
- Understand label switching, forwarding, and load balancing.

Restrictions for MPLS EM—MPLS LSP Multipath Tree Trace

- All restrictions that apply to the MPLS LSP Ping and LSP Traceroute features also apply to the MPLS EM—MPLS LSP Multipath Tree Trace feature:
 - You cannot use the MPLS LSP Multipath Tree Trace feature to trace the path taken by AToM packets. The MPLS LSP Multipath Tree Trace feature is not supported for AToM. (MPLS LSP Ping is supported for AToM.) However, you can use the MPLS LSP Multipath Tree Trace feature to troubleshoot the Interior Gateway Protocol (IGP) LSP that is used by AToM.
 - You cannot use the MPLS LSP Multipath Tree Trace feature to validate or trace MPLS Virtual Private Networks (VPNs). Multiple LSP paths are not discovered unless all routers in the MPLS core support an RFC 4379 implementation of *Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures*.
- MPLS LSP multipath tree trace is not expected to operate in networks that support time-to-live (TTL) hiding.

Information About MPLS EM—MPLS LSP Multipath Tree Trace

Before using the MPLS EM—MPLS LSP Multipath Tree Trace feature, you need an understanding of the following concepts:

- [Overview of MPLS LSP Multipath Tree Trace, page 3](#)
- [Discovery of IPv4 Load Balancing Paths by MPLS LSP Multipath Tree Trace, page 3](#)
- [Echo Reply Return Codes Sent by the Router Processing Multipath LSP Tree Trace, page 4](#)

Overview of MPLS LSP Multipath Tree Trace

As the number of MPLS deployments increases, the number of traffic types the MPLS networks carry could increase. In addition, load balancing on label switch routers (LSRs) in the MPLS network provides alternate paths for carrying MPLS traffic to a target router. The ability of service providers to monitor LSPs and quickly isolate MPLS forwarding problems is critical to their ability to offer services.

Prior to the release of the MPLS EM—MPLS LSP Multipath Tree Trace feature no automated way existed to discover all paths between provider edge (PE) routers. Troubleshooting forwarding problems between PEs was cumbersome.

The release of the MPLS EM—MPLS LSP Multipath Tree Trace feature provides an automated way to discover all paths from the ingress PE router to the egress PE router in multivendor networks that use IPv4 load balancing at the transit routers. Once the PE-to-PE paths are discovered, use MPLS LSP ping and MPLS LSP traceroute to periodically test them.

The MPLS EM—MPLS LSP Multipath Tree Trace feature requires the Cisco RFC-compliant implementation that is based on RFC 4379. If you do not have a Cisco IOS release that supports RFC 379, MPLS LSP multipath tree trace does not operate to discover all PE-to-PE paths.

Discovery of IPv4 Load Balancing Paths by MPLS LSP Multipath Tree Trace

IPv4 load balancing at a transit router is based on the incoming label stack and the source and destination addresses in the IP header. The outgoing label stack and IP header source address remain constant for each branch being traced.

When you execute MPLS LSP multipath tree trace on the source LSR, the router needs to find the set of IP header destination addresses to use all possible output paths. The source LSR starts path discovery by sending a transit router a bitmap in an MPLS echo request. The transit router returns information in an MPLS echo request that contains subsets of the bitmap in a downstream map (DS Map) in an echo reply. The source router can then use the information in the echo reply to interrogate the next router. The source router interrogates each successive router until it finds one bitmap setting that is common to all routers along the path. The router uses TTL expiry to interrogate the routers to find the common bits.

For example, you could start path discovery by entering the following command at the source router:

```
Router# trace mpls multipath ipv4 10.131.101.129/32 hashkey ipv4 bitmap 16
```

This command sets the IP address of the target router as 10.131.101.192 255.255.255.255 and configures:

- The default hash key type to 8, which requests that an IPv4 address prefix and bit mask address set be returned in the DS Map in the echo reply.

- The bitmap size to 16. This means that MPLS LSP multipath tree trace uses 16 addresses (starting with 127.0.0.1) in the discovery of all paths of an LSP between the source router and the target router.

If you enter the `trace mpls multipath ipv4 10.131.101.129/32` command, MPLS LSP multipath tree trace uses the default hash type of 8 or IPv4 and a default bitmap size of 32. Your choice of a bitmap size depends on the number of routes in your network. If you have a large number of routes, you might need to use a larger bitmap size.

Echo Reply Return Codes Sent by the Router Processing Multipath LSP Tree Trace

[Table 1](#) describes the characters that the router processing a multipath LSP tree trace packet returns to the sender about the failure or success of the request.

Table 1 *Echo Reply Return Codes*

Output Code	Echo Return Code	Meaning
Period “.”	—	A timeout occurred before the target router could reply.
x	0	No return code.
M	1	Malformed request.
m	2	Unsupported type, length, values (TLVs).
!	3	Success.
F	4	No Forwarding Equivalence Class (FEC) mapping.
D	5	DS Map mismatch.
R	6	Downstream router but not target.
U	7	Reserved.
L	8	Labeled output interface.
B	9	Unlabeled output interface.
f	10	FEC mismatch.
N	11	No label entry.
P	12	No receive interface label protocol.
p	13	Premature termination of the LSP.
X	unknown	Undefined return code.

How to Configure MPLS EM—MPLS LSP Multipath Tree Trace

This section contains the following tasks:

- [Customizing the Default Behavior of MPLS Echo Packets, page 5](#) (optional)
- [Configuring MPLS LSP Multipath Tree Trace, page 7](#) (required)
- [Discovering IPv4 Load Balancing Paths Using MPLS LSP Multipath Tree Trace, page 9](#) (required)

- [Monitoring LSP Paths Discovered by MPLS LSP Multipath Tree Trace Using MPLS LSP Traceroute, page 10 \(optional\)](#)
- [Using DSCP to Request a Specific Class of Service in an Echo Reply, page 13 \(optional\)](#)
- [Controlling How a Responding Router Replies to an MPLS Echo Request, page 14 \(optional\)](#)
- [Specifying the Output Interface for Echo Packets Leaving a Router for MPLS LSP Multipath Tree Trace, page 16 \(optional\)](#)
- [Setting the Pace of MPLS Echo Request Packet Transmission for MPLS LSP Multipath Tree Trace, page 17 \(optional\)](#)
- [Enabling MPLS LSP Multipath Tree Trace to Detect LSP Breakages Caused by an Interface That Lacks an MPLS Configuration, page 18 \(optional\)](#)
- [Requesting That a Transit Router Validate the Target FEC Stack for MPLS LSP Multipath Tree Trace, page 19 \(optional\)](#)
- [Setting the Number of Timeout Attempts for MPLS LSP Multipath Tree Trace, page 21 \(optional\)](#)

Customizing the Default Behavior of MPLS Echo Packets

Perform the following task to customize the default behavior of MPLS echo packets. You might need to customize the default echo packet encoding and decoding behavior to allow later implementations of the *Detecting MPLS Data Plane Failures* (RFC 4379) to be deployed in networks running earlier versions of the draft.

MPLS Embedded Management Configuration

Before using the **ping mpls**, **trace mpls**, or **trace mpls multipath** command, you should consider ensuring that the router is configured to encode and decode MPLS echo packets in a format that all receiving routers in the network can understand.

LSP ping drafts after Version 3 (draft-ietf-mpls-ping-03) have undergone numerous TLV format changes, but the implementations based on different drafts might not interoperate properly.

To allow later Cisco implementations to interoperate with draft Version 3 Cisco and non-Cisco implementations, a global configuration mode (MPLS OAM configuration) allows you to encode and decode echo packets in formats specified by draft Version 3 implementations.

Unless configured otherwise, a Cisco implementation encodes and decodes echo requests assuming the version on which the Internet Engineering Task Force (IETF) implementation is based.

To allow for seamless interoperability with earlier Revision 1 and 3 images, you can use MPLS Operation, Administration, and Maintenance (OAM) configuration mode parameters to force the default behavior of the Revision 4 images to be compliant or compatible in networks with Revision 1 or Revision 3 images.

To prevent failures reported by the replying router due to TLV version issues, you should configure all routers in the core. Encode and decode MPLS echo packets in the same draft version. For example, if the network is running RFC 4379 (Cisco Revision 4) implementations but one router is capable of only Version 3 (Cisco Revision 3), configure all routers in the network to operate in Revision 3 mode.

Cisco Revision 4 is the default version. The default version is the latest LSP Ping version supported by the image on the router.

Prerequisites

MPLS LSP Multipath Tree Trace requires RFC 4379 (Revision 4).

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mpls oam**
4. **echo revision {3 | 4}**
5. **[no] echo vendor-extension**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	mpls oam Example: Router(config)# mpls oam	Enters MPLS OAM configuration mode and customizes the default behavior of echo packets.
Step 4	echo revision {3 4} Example: Router(config-mpls)# echo revision 4	Customizes the default behavior of echo packets. <ul style="list-style-type: none"> • The revision keyword set echo packet attributes to one of the following: <ul style="list-style-type: none"> – 3 = draft-ietf-mpls-ping-03 (Revision 2) – 4 = RFC 4379 compliant (default) <p>Note The MPLS LSP Multipath Tree Trace feature requires Revision 4.</p>

	Command or Action	Purpose
Step 5	<pre>[no] echo vendor-extension</pre> <p>Example: <pre>Router(config-mpls)# echo vendor-extension</pre></p>	<p>Customizes the default behavior of echo packets.</p> <ul style="list-style-type: none"> The vendor-extension keyword sends the Cisco-specific extension of TLVs with the echo packets. The no form of the command allows you to disable a Cisco vendor's extension TLVs that another vendor's noncompliant implementations may not support. <p>The router default is echo vendor-extension.</p>
Step 6	<pre>end</pre> <p>Example: <pre>Router(config-mpls)# end</pre></p>	<p>Exits to privileged EXEC mode.</p>

Configuring MPLS LSP Multipath Tree Trace

Perform the following task to configure MPLS multipath LSP traceroute. This task helps discover all LSPs from an egress router to an ingress router.

Prerequisites

Cisco LSP ping or traceroute implementations based on draft-ietf-mpls-lsp-ping-11 are capable in some cases of detecting the formatting of the sender of an MPLS echo request. However, certain cases exist in which an echo request or echo reply might not contain the Cisco extension TLV. To avoid complications due to certain cases where the echo packets are decoded assuming the wrong TLV formats, configure all routers in the network to operate in the same mode.

For an MPLS LSP multipath tree trace to be successful, the implementation in your routers must support RFC 4379 on all core routers.

If all routers in the network support RFC-4379 and another vendor's implementation exists that is not capable of properly handling Cisco's vendor TLV, the routers supporting the RFC-compliant or later configuration must include commands to disable the Cisco vendor TLV extensions.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mpls oam**
4. **echo revision 4**
5. **[no] echo vendor-extension**
6. **end**
7. **trace mpls multipath ipv4 destination-ip-address/destination-mask-length**
8. **debug mpls lspv multipath**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example: Router> enable</p>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example: Router# configure terminal</p>	<p>Enters global configuration mode.</p>
Step 3	<p>mpls oam</p> <p>Example: Router(config)# mpls oam</p>	<p>Enters MPLS OAM configuration mode.</p>
Step 4	<p>echo revision 4</p> <p>Example: Router(config-mpls)# echo revision 4</p>	<p>Customizes the default behavior of echo packets.</p> <ul style="list-style-type: none"> The revision 4 keywords set echo packet attributes to the default Revision 4 (RFC 4379 compliant). <p>Note The MPLS LSP Multipath Tree Trace feature requires Revision 4.</p>
Step 5	<p>[no] echo vendor-extension</p> <p>Example: Router(config-mpls) echo vendor-extension</p>	<p>(Optional) Customizes the default behavior of echo packets.</p> <ul style="list-style-type: none"> The vendor-extension keyword sends the Cisco-specific extension of TLVs with the echo packets. The no form of the command allows you to disable a Cisco vendor's extension TLVs that another vendor's noncompliant implementations may not support. <p>The router default is echo vendor-extension.</p>
Step 6	<p>end</p> <p>Example: Router(config-mpls)# end</p>	<p>Exits to privileged EXEC mode.</p>
Step 7	<p>trace mpls multipath ipv4 <i>destination-ip-address/destination-mask-length</i></p> <p>Example: Router# trace mpls multipath ipv4 10.131.161.251/32</p>	<p>Discovers all LSPs from an egress router to an ingress router.</p> <ul style="list-style-type: none"> The ipv4 keyword specifies the destination type as an LDP IPv4 address. The <i>destination-ip-address</i> argument is the address prefix of the target to be tested. The <i>destination-mask-length</i> argument is the number of bits in the network mask of the target address. The / keyword before this argument is required.
Step 8	<p>debug mpls lspv multipath</p> <p>Example: Router# debug mpls lspv multipath</p>	<p>Displays multipath information related to the MPLS LSP Multipath Tree Trace feature.</p>

Discovering IPv4 Load Balancing Paths Using MPLS LSP Multipath Tree Trace

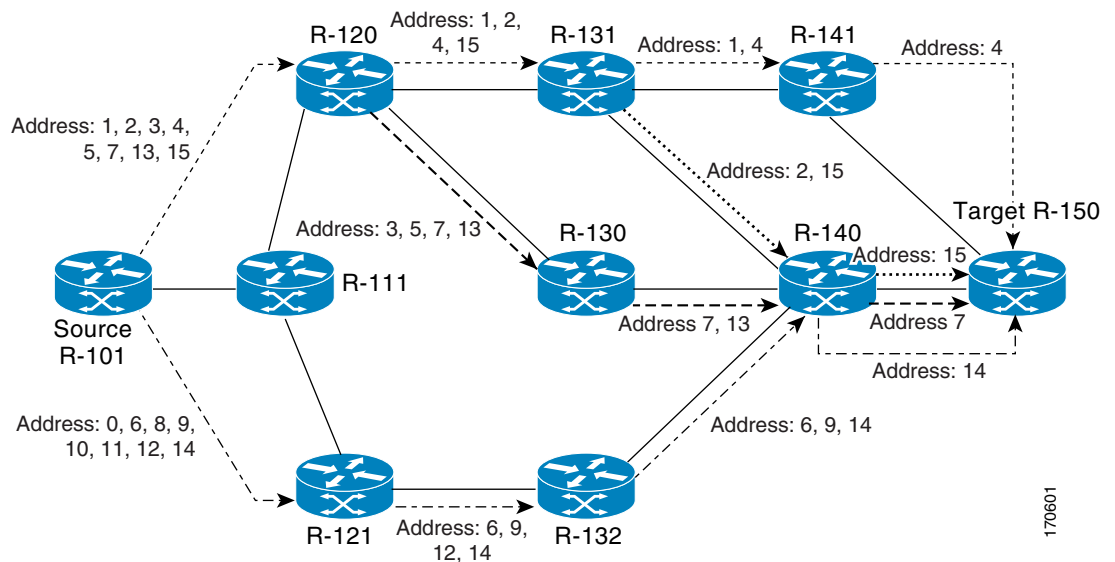
Perform the following task to discover IPv4 load balancing paths using MPLS LSP multipath tree trace.

MPLS Multipath LSP Traceroute Path Discovery

A Cisco router load balances MPLS packets based on the incoming label stack and the source and destination addresses in the IP header. The outgoing label stack and IP header source address remain constant for each path being traced. The router needs to find the set of IP header destination addresses to use all possible output paths. This might require exhaustive searching of the 127.x.y.z/8 address space. Once you discover all paths from the source LSR to the target or destination LSR with MPLS LSP multipath tree trace, you can use MPLS LSP traceroute to monitor these paths.

Figure 1 shows how MPLS LSP multipath tree trace discovers LSP paths in a sample network. In Figure 1, the bitmap size is 16 and the numbers 0 to 15 represent the bitmapped addresses that MPLS LSP multipath tree trace uses to discover all the paths from the source LSR R-101 to the target LSR R-150. Figure 1 illustrates how the `trace mpls multipath` command discovers all LSP paths in the sample network.

Figure 1 MPLS LSP Multipath Tree Trace Path Discovery in a Sample Network



SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `mpls oam`
4. `echo revision 4`
5. `end`
6. `trace mpls multipath ipv4 destination-ip-address/destination-mask-length hashkey ipv4 bitmap bitmap-size`

DETAILED STEPS

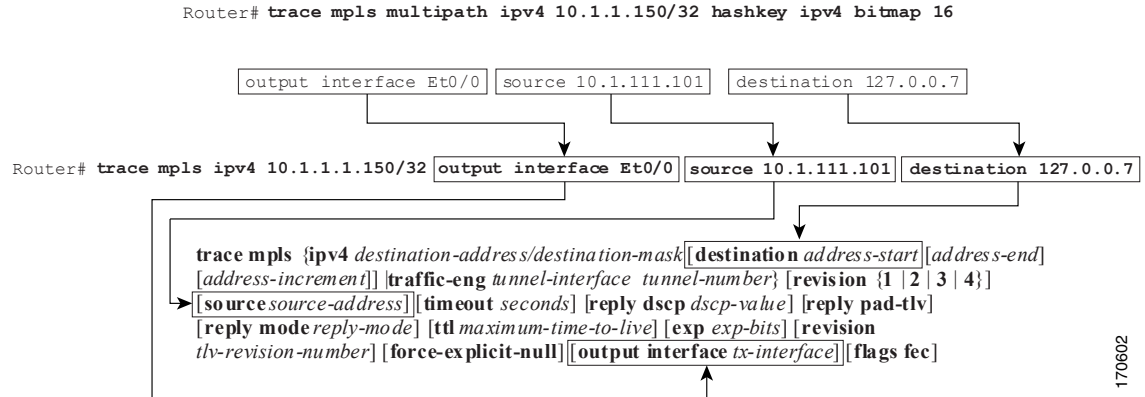
	Command or Action	Purpose
Step 1	<code>enable</code> Example: Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	<code>configure terminal</code> Example: Router# configure terminal	Enters global configuration mode.
Step 3	<code>mpls oam</code> Example: Router(config)# mpls oam	Enters MPLS OAM configuration mode and sets the echo packet attribute to Revision 4 (RFC 4379 compliant).
Step 4	<code>echo revision 4</code> Example: Router(config-mpls)# echo revision 4	Customizes the default behavior of echo packets. <ul style="list-style-type: none"> The revision 4 keywords set echo packet attributes to the default Revision 4 (RFC 4379 compliant). Note The MPLS LSP Multipath Tree Trace feature requires Revision 4.
Step 5	<code>end</code> Example: Router(config-mpls)# end	Exits to privileged EXEC mode.
Step 6	<code>trace mpls multipath ipv4 destination-address/destination-mask-length hashkey ipv4 bitmap bitmap-size</code> Example: Router# trace mpls multipath ipv4 10.131.161.251/32 hashkey ipv4 bitmap 16	Discovers all MPLS LSPs from an egress router to an ingress router. <ul style="list-style-type: none"> The ipv4 keyword specifies the destination type as an LDP IPv4 address. The <i>destination-address</i> argument is the address prefix of the target to be tested. The <i>destination-mask-length</i> argument is the number of bits in the network mask of the target address. The <i>/</i> keyword before this argument is required. The hashkey ipv4 keywords set the hashkey type to IPv4 addresses. The bitmap bitmap-size keyword and arguments set the bitmap size for multipath discovery.

Monitoring LSP Paths Discovered by MPLS LSP Multipath Tree Trace Using MPLS LSP Traceroute

Perform the following task to monitor LSP paths discovered by MPLS LSP multipath tree trace using MPLS LSP traceroute. You can take output directly from the **trace mpls multipath** command and add it to a **trace mpls** command periodically to verify that the path is still operating.

Figure 2 shows the mapping of the output of a `trace mpls multipath` command to a `trace mpls` command.

Figure 2 Mapping of trace mpls multipath Command Output to a trace mpls Command



Each path you discover with MPLS LSP Multipath Tree Trace can be tested in this manner periodically to monitor the LSP paths in your network.

SUMMARY STEPS

1. **enable**
2. **trace mpls multipath ipv4** *destination-address/destination-mask-length* **hashkey ipv4** *bitmap* *bitmap-size*
3. **trace mpls ipv4** *destination-address/destination-mask-length* [**output interface** *tx-interface*] [**source** *source-address*] [**destination** *address-start*]
4. **exit**

DETAILED STEPS

Step 1 enable

Use this command to enable privileged EXEC mode. Enter your password if prompted. For example:

```
Router> enable
Router#
```

Step 2 trace mpls multipath ipv4 destination-address/destination-mask-length hashkey ipv4 bitmap bitmap-size

Use this command to discover all MPLS LSPs from an egress router to an ingress router. For example:

```
Router# trace mpls multipath ipv4 10.1.1.150/32 hashkey ipv4 bitmap 16
```

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 468 ms

```

The output of the **trace mpls multipath** command in the example shows the result of path discovery with MPLS LSP multipath tree trace. In this example, the command sets the bitmap size to 16. Path discovery starts by MPLS LSP multipath tree trace using 16 bitmapped addresses as it locates LSP paths from the source router to the target router with prefix and mask 10.1.1.150/32. MPLS LSP multipath tree trace starts using the 127.x.y.z/8 address space with 127.0.0.1.

Step 3 **trace mpls ipv4 destination-address/destination-mask-length [output interface tx-interface] [source source-address] [destination address-start]**

Use this command to verify that the paths discovered when you entered a **trace mpls multipath** command are still operating. For example, the output for Path 0 in the previous **trace mpls multipath** command in [Step 2](#) is:

```
output interface Et0/0 source 10.1.111.101 destination 127.0.0.0
```

If you put the output for path 0 in the **trace mpls** command, you see the following results:

```
Router# trace mpls ipv4 10.1.1.150/32 output interface Et0/0 source 10.1.111.101 destination 127.0.0.0
```

```
Tracing MPLS Label Switched Path to 10.1.1.150/32, 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 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.
 0 10.1.111.101 MRU 1500 [Labels: 33 Exp: 0]
L 1 10.1.111.111 MRU 1500 [Labels: 34 Exp: 0] 40 ms
L 2 10.2.121.121 MRU 1500 [Labels: 34 Exp: 0] 32 ms
L 3 10.3.132.132 MRU 1500 [Labels: 32 Exp: 0] 16 ms
L 4 10.4.140.240 MRU 1504 [Labels: implicit-null Exp: 0] 20 ms
! 5 10.5.150.50 20 ms

```

You can take output directly from the **trace mpls multipath** command and add it to a **trace mpls** command periodically to verify that the path is still operating (see [Figure 2](#)).

Step 4 **exit**

Use this command to exit to user EXEC mode. for example:

```
Router# exit  
Router>
```

Using DSCP to Request a Specific Class of Service in an Echo Reply

For Cisco IOS Release 12.2(27)SXE, Cisco added a reply differentiated services code point (DSCP) option that lets you request a specific class of service (CoS) in an echo reply.

The reply DSCP option is supported in the experimental mode for IETF draft-ietf-mpls-lsp-ping-03.txt. Cisco implemented a vendor-specific extension for the reply DSCP option rather than using a Reply TOS TLV. A Reply TOS TLV serves the same purpose as the **reply dscp** command in IETF draft-ietf-mpls-lsp-ping-11.txt. This draft provides a standardized method of controlling the reply DSCP.

**Note**

Before RFC 4379, Cisco implemented the Reply DSCP option as an experimental capability using a Cisco vendor extension TLV. If a router is configured to encode MPLS echo packets for draft Version 3 implementations, a Cisco vendor extension TLV is used instead of the = Reply TOS TLV that was defined in draft Version 8.

To use DSCP to request a specific CoS in an echo reply, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **trace mpls multipath ipv4** *destination-address/destination-mask-length* [**reply dscp** *dscp-value*]
3. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<pre>enable</pre> <p>Example: Router> enable</p>	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	<pre>trace mpls multipath ipv4 destination-address/ destination-mask-length [reply dscp dscp-value]</pre> <p>Example: Router# trace mpls multipath ipv4 10.131.191.252/32 reply dscp 50</p>	Discovers all MPLS LSPs from an ingress router to an egress router and controls the DSCP value of an echo reply. <ul style="list-style-type: none"> The ipv4 keyword specifies the destination type as an LDP IPv4 address. The <i>destination-address</i> argument is the address prefix of the target to be tested. The <i>destination-mask-length</i> argument is the number of bits in the network mask of the target address. The <i>/</i> keyword before this argument is required. The reply dscp dscp-value keywords and argument are the DSCP value of an echo reply. A Reply TOS TLV serves the same purpose as the reply dscp command in IETF draft-ietf-mpls-lsp-ping-11.txt. <p>Note To specify a DSCP value, you must enter the reply dscp dscp-value keywords and argument.</p>
Step 3	<pre>exit</pre> <p>Example: Router# exit</p>	Returns to user EXEC mode.

Controlling How a Responding Router Replies to an MPLS Echo Request

This section contains information about and instructions for controlling how a responding router replies to an MPLS echo request. You should understand the following information before you configure a reply mode for the echo request response:

- [Reply Modes for an MPLS LSP Multipath Tree Trace Echo Request Response, page 14](#)

Reply Modes for an MPLS LSP Multipath Tree Trace Echo Request Response

The reply mode controls how a responding router replies to an MPLS echo request sent by a **trace mpls multipath** command. There are two reply modes for an echo request packet:

- ipv4—Reply with an IPv4 User Datagram Protocol (UDP) packet (default)
- router-alert—Reply with an IPv4 UDP packet with router alert



Note

Use the ipv4 and router-alert reply modes with each other to prevent false negatives. If you do not receive a reply via the ipv4 mode, send a test with the router-alert reply mode. If both fail, something is wrong in the return path. The problem might be due to an incorrect ToS setting.

IPv4 UDP Reply Mode

The IPv4 UDP reply mode is the most common reply mode used with a **trace mpls multipath** command when you want to periodically poll the integrity of an LSP. With this option, you do not have explicit control over whether the packet traverses IP or MPLS hops to reach the originator of the MPLS echo request. If the originating (headend) router fails to receive a reply to an MPLS echo request when you use the **reply mode ipv4** keywords, use the **reply mode router-alert** keywords.

Router-alert Reply Mode

The router-alert reply mode adds the router alert option to the IP header. When an IP packet that contains an IP router alert option in its IP header or an MPLS packet with a router alert label as its outermost label arrives at a router, the router punts (redirects) the packet to the Route Processor (RP) process level for handling. This forces the RP of each intermediate router to specifically handle the packet at each intermediate hop as it moves back to the destination. Hardware and line-card forwarding inconsistencies are thus bypassed. Router-alert reply mode is slower than IPv4 mode because the reply requires process-level RP handling at each hop.

[Table 2](#) describes how an incoming IP packet with an IP router alert is handled by the router switching path processes when the outgoing packet is an IP packet or an MPLS packet. It also describes how an MPLS packet with a router alert option is handled by the router switching path processes when the outgoing packet is an IP packet or an MPLS packet.

Table 2 Path Process Handling of IP and MPLS Router Alert Packets

Incoming Packet	Outgoing Packet	Normal Switching Action	Process Switching Action
IP packet—Router alert option in IP header	IP packet—Router alert option in IP header	Router alert option in IP header causes the packet to be punted to the process switching path.	Forwards the packet as is
	MPLS packet		Forwards the packet as is
MPLS packet—Outermost label contains a router alert	IP packet—Router alert option in IP header	If the router alert label is the outermost label, it causes the packet to be punted to the process switching path.	Removes the outermost router alert label and forwards the packet as an IP packet
	MPLS packet—Outermost label contains a router alert		Preserves the outermost router alert label and forwards the MPLS packet

SUMMARY STEPS

1. **enable**
2. **trace mpls multipath ipv4** *destination-address/destination-mask-length* **reply mode** {**ipv4** | **router-alert**}
3. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<pre>enable</pre> <p>Example: Router> enable </p>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	<pre>trace mpls multipath ipv4 destination-address/ destination-mask-length reply mode {ipv4 router-alert}</pre> <p>Example: Router# trace mpls multipath ipv4 10.131.191.252/32 reply mode router-alert </p>	<p>Discovers all MPLS LSPs from an ingress router to an egress router and specifies the reply mode.</p> <ul style="list-style-type: none"> The ipv4 keyword specifies the destination type as an LDP IPv4 address. The <i>destination-address</i> argument is the address prefix of the target to be tested. The <i>destination-mask-length</i> argument is the number of bits in the network mask of the target address. The <i>/</i> keyword before this argument is required. The reply mode keyword requires that you enter one of the following keywords to specify the reply mode: <ul style="list-style-type: none"> The ipv4 keyword—Reply with an IPv4 UDP packet (default). The router-alert keyword—Reply with an IPv4 UDP packet with router alert. <p>Note To specify the reply mode, you must enter the reply mode keyword with the ipv4 or router-alert keyword.</p>
Step 3	<pre>exit</pre> <p>Example: Router# exit </p>	<p>Returns to user EXEC mode.</p>

Specifying the Output Interface for Echo Packets Leaving a Router for MPLS LSP Multipath Tree Trace

Perform the following task to specify the output interface for echo packets leaving a router for the MPLS LSP Multipath Tree Trace feature. You can use this task to test the LSPs reachable through a given interface.

Echo Request Output Interface Control

You can control the interface through which packets leave a router. Path output information is used as input to LSP ping and traceroute.

The echo request output interface control feature allows you to force echo packets through the paths that perform detailed debugging or characterizing of the LSP. This feature is useful if a PE router connects to an MPLS cloud and there are broken links. You can direct traffic through a certain link. The feature also is helpful for troubleshooting network problems.

SUMMARY STEPS

1. **enable**
2. **trace mpls multipath ipv4 destination-address/destination-mask-length [output interface tx-interface]**
3. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	trace mpls multipath ipv4 destination-address/destination-mask-length [output interface tx-interface] Example: Router# trace mpls multipath ipv4 10.131.159.251/32 output interface ethernet0/0	Discovers all MPLS LSPs from an ingress router to an egress router and specifies the interface through which echo packets leave a router. <ul style="list-style-type: none"> • The ipv4 keyword specifies the destination type as an LDP IPv4 address. • The <i>destination-address</i> argument is the address prefix of the target to be tested. • The <i>destination-mask-length</i> argument is the number of bits in the network mask of the target address. The <i>/</i> keyword before this argument is required. • The output interface tx-interface keywords and argument specify the output interface for the MPLS echo request. <p>Note You must specify the output interface keywords.</p>
Step 3	exit Example: Router# exit	Returns to user EXEC mode.

Setting the Pace of MPLS Echo Request Packet Transmission for MPLS LSP Multipath Tree Trace

Perform the following task to set the pace of MPLS echo request packet transmission for the MPLS LSP Multipath Tree Trace feature. Echo request traffic pacing allows you to set the pace of the transmission of packets so that the receiving router does not drop packets. If you have a large amount of traffic on your network you might increase the size of the interval to help ensure that the receiving router does not drop packets.

SUMMARY STEPS

1. **enable**
2. **trace mpls multipath ipv4 destination-address/destination-mask-length [interval milliseconds]**

3. exit

DETAILED STEPS

	Command or Action	Purpose
Step 1	<pre>enable</pre> <p>Example: Router> enable </p>	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	<pre>trace mpls multipath ipv4 destination-address/ destination-mask-length [interval milliseconds]</pre> <p>Example: Router# trace mpls multipath ipv4 10.131.159.251/32 interval 100 </p>	Discovers all MPLS LSPs from an egress router to an ingress router and sets the time in milliseconds between successive MPLS echo requests. <ul style="list-style-type: none"> The ipv4 keyword specifies the destination type as an LDP IPv4 address. The <i>destination-address</i> argument is the address prefix of the target to be tested. The <i>destination-mask</i> argument is the number of bits in the network mask of the target address. The / keyword before this argument is required. The interval milliseconds keyword and argument set the time between successive MPLS echo requests in milliseconds. The default is 0 milliseconds. <p>Note To pace the transmission of packets, you must specify the interval keyword.</p>
Step 3	<pre>exit</pre> <p>Example: Router# exit </p>	Returns to user EXEC mode.

Enabling MPLS LSP Multipath Tree Trace to Detect LSP Breakages Caused by an Interface That Lacks an MPLS Configuration

Perform the following task to enable MPLS LSP multipath tree trace to detect LSP breakages caused by an interface that lacks an MPLS configuration. If an interface is not configured for MPLS, then it cannot forward MPLS packets.

Explicit Null Label Shimming Tests LSP Ability to Carry MPLS Traffic

For an MPLS LSP multipath tree trace of LSPs carrying IPv4 FECs, you can force an explicit null label to be added to the MPLS label stack even though the label was unsolicited. This allows MPLS LSP multipath tree trace to detect LSP breakages caused by an interface that is not configured for MPLS. MPLS LSP multipath tree trace does not report that an LSP is functioning when it is unable to send MPLS traffic.

An explicit null label is added to an MPLS label stack if MPLS echo request packets are forwarded from an interface not configured for MPLS that is directly connected to the destination of the MPLS LSP multipath tree trace or if the IP TTL value for the MPLS echo request packets is set to 1.

When you enter a **trace mpls multipath** command, you are looking for all MPLS LSP paths from an egress router to an ingress router. Failure at output interfaces that are not configured for MPLS at the penultimate hop are not detected. Explicit-null shimming allows you to test an LSP's ability to carry MPLS traffic.

SUMMARY STEPS

1. **enable**
2. **trace mpls multipath ipv4 destination-address/destination-mask-length force-explicit-null**
3. **exit**

DETAILED STEP

Step 1 enable Example: Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2 trace mpls multipath ipv4 destination-address/destination-mask-length force-explicit-null Example: Router# trace mpls multipath ipv4 10.131.191.252/32 force-explicit-null	Discovers all MPLS LSPs from an egress router to an ingress router and forces an explicit null label to be added to the MPLS label stack. <ul style="list-style-type: none"> • The ipv4 keyword specifies the destination type as an LDP IPv4 address. • The <i>destination-address</i> argument is the address prefix of the target to be tested. • The <i>destination-mask-length</i> argument is the number of bits in the network mask of the target address. The <i>/</i> keyword before this argument is required. • The force-explicit-null keyword forces an explicit null label to be added to the MPLS label stack even though the label was unsolicited. <p>Note You must enter the force-explicit-null keyword to enable MPLS LSP multipath tree trace to detect LSP breakages caused by an interface that is not configured for MPLS.</p>
Step 3 exit Example: Router# exit	Returns to user EXEC mode.

Requesting That a Transit Router Validate the Target FEC Stack for MPLS LSP Multipath Tree Trace

Perform the following task to request that a transit router validate the target FEC stack for the MPLS LSP Multipath Tree Trace feature.

An MPLS echo request tests a particular LSP. The LSP to be tested is identified by the FEC stack.

During an MPLS LSP multipath tree trace, the echo packet validation rules do not require that a transit router validate the target FEC stack TLV. A downstream map TLV containing the correct received labels must be present in the echo request for target FEC stack checking to be performed.

To request that a transit router validate the target FEC stack, set the V flag from the source router by entering the **flags fec** keywords in the **trace mpls multipath** command. The default is that echo request packets are sent with the V flag set to 0.

SUMMARY STEPS

1. **enable**
2. **trace mpls multipath ipv4** *destination-address/destination-mask-length* [**flags fec**] [**ttl** *maximum-time-to-live*]
3. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	trace mpls multipath ipv4 <i>destination-address/destination-mask-length</i> [flags fec] [ttl <i>maximum-time-to-live</i>] Example: Router# trace mpls multipath ipv4 10.131.159.252/32 flags fec ttl 5	Discovers all MPLS LSPs from an egress router to an ingress router and requests validation of the target FEC stack by a transit router. <ul style="list-style-type: none"> • The ipv4 keyword specifies the destination type as an LDP IPv4 address. • The <i>destination-address</i> argument is the address prefix of the target to be tested. • The <i>destination-mask-length</i> argument is the number of bits in the network mask of the target address. The <i>/</i> keyword before this argument is required. • The flags fec keywords requests that target FEC stack validation be done at a transit router. • The ttl <i>maximum-time-to-live</i> keyword and argument pair specify a maximum hop count. <p>Note For a transit router to validate the target FEC stack, you must enter the flags fec and ttl keywords.</p>
Step 3	exit Example: Router# exit	Returns to user EXEC mode.

Setting the Number of Timeout Attempts for MPLS LSP Multipath Tree Trace

Perform the following task to set the number of timeout attempts for the MPLS LSP Multipath Tree Trace feature.

A retry is attempted if an outstanding echo request times out waiting for the corresponding echo reply.

SUMMARY STEPS

1. **enable**
2. **trace mpls multipath ipv4** *destination-address/destination-mask-length* [**retry-count** *retry-count-value*]
3. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	trace mpls multipath ipv4 <i>destination-address/destination-mask-length</i> [retry-count <i>retry-count-value</i>] Example: Router# trace mpls multipath ipv4 10.131.159.252/32 retry-count 4	Sets the number of retry attempts during an MPLS LSP multipath tree trace. <ul style="list-style-type: none"> • The ipv4 keyword specifies the destination type as an LDP IPv4 address. • The <i>destination-address</i> argument is the address prefix of the target to be tested. • The <i>destination-mask-length</i> argument is the number of bits in the network mask of the target address. The <i>/</i> keyword before this argument is required. • The retry-count <i>retry-count-value</i> keyword and argument sets the number of retry attempts after a timeout occurs. <p>A retry-count value of “0” means infinite retries. A retry-count value from 0 to 10 is suggested. You might want to increase the retry value to greater than 10, if 10 is too small a value. The default retry-count value is 3.</p> <p>Note To set the number of retries after a timeout, you must enter the retry-count keyword.</p>
Step 3	exit Example: Router# exit	Returns to user EXEC mode.

Configuration Examples for MPLS EM—MPLS LSP Multipath Tree Trace

This section includes the following configuration examples for the MPLS EM—MPLS LSP Multipath Tree Trace feature:

- [Customizing the Default Behavior of MPLS Echo Packets: Example, page 22](#)
- [Configuring MPLS LSP Multipath Tree Trace: Example, page 22](#)
- [Using DSCP to Request a Specific Class of Service in an Echo Reply: Example, page 24](#)
- [Controlling How a Responding Router Replies to an MPLS Echo Request: Example, page 25](#)
- [Specifying the Output Interface for Echo Packets Leaving a Router for MPLS LSP Multipath Tree Trace: Example, page 25](#)
- [Setting the Pace of MPLS Echo Request Packet Transmission for MPLS LSP Multipath Tree Trace: Example, page 26](#)
- [Enabling MPLS LSP Multipath Tree Trace to Detect LSP Breakages Caused by an Interface That Lacks an MPLS Configuration: Example, page 27](#)
- [Requesting That a Transit Router Validate the Target FEC Stack for MPLS LSP Multipath Tree Trace: Example, page 29](#)
- [Setting the Number of Timeout Attempts for MPLS LSP Multipath Tree Trace: Example, page 29](#)

Customizing the Default Behavior of MPLS Echo Packets: Example

The following example shows how to customize the behavior of MPLS echo packets so that the MPLS LSP Multipath Tree Trace feature interoperates with a vendor implementation that does not interpret RFC 4379 as Cisco does:

```
configure terminal
!
mpls oam
 echo revision 4
no echo vendor-extension
end
```

The **echo revision** command is included in this example for completeness. The default echo revision number is 4, which corresponds to RFC 4379.

Configuring MPLS LSP Multipath Tree Trace: Example

The following example shows how to configure the MPLS LSP Multipath Tree Trace feature to interoperate with a vendor implementation that does not interpret RFC 4379 as Cisco does:

```
configure terminal
!
mpls oam
 echo revision 4
no echo vendor-extension
end
!
trace mpls multipath ipv4 10.131.161.151/32
```

The **echo revision** command is included in this example for completeness. The default echo revision number is 4, which corresponds to the RFC 4379.

Discovering IPv4 Load Balancing Paths Using MPLS LSP Multipath Tree Trace: Example

The following example shows how to use the MPLS LSP Multipath Tree Trace feature to discover IPv4 load balancing paths. The example is based on the sample network shown in [Figure 3](#). In this example, the bitmap size is set to 16. Therefore, path discovery starts by the MPLS LSP Multipath Tree Trace feature using 16 bitmapped addresses as it locates LSP paths from the source router R-101 to the target router R-150 with prefix and mask 10.1.1.150/32. The MPLS LSP Multipath Tree Trace feature starts using the 127.x.y.z/8 address space with 127.0.0.0.

```
Router# trace mpls multipath ipv4 10.1.1.150/32 hashkey ipv4 bitmap 16
```

```
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)
```

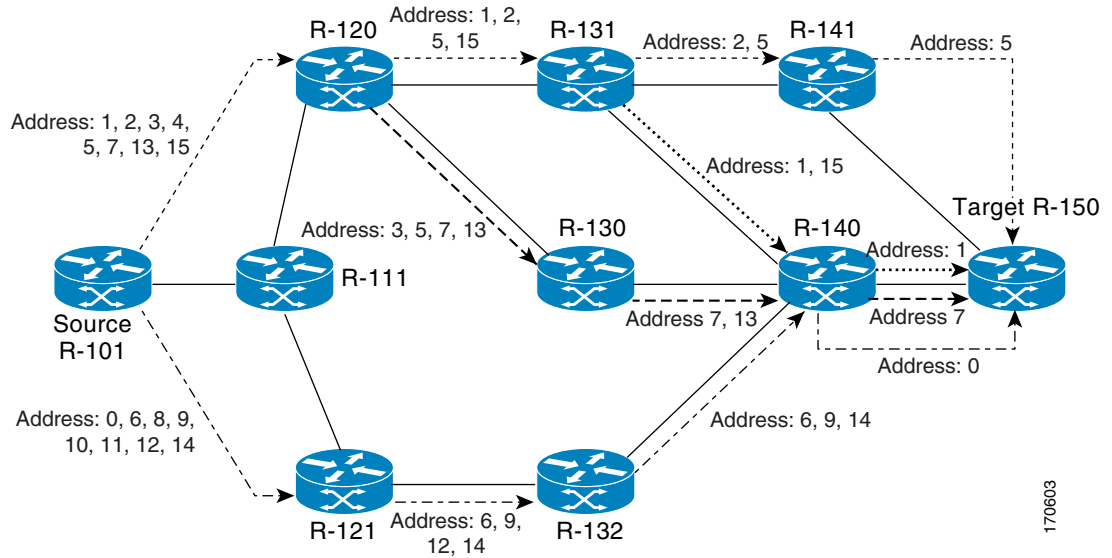
```
Echo Request (sent/fail) (14/0)
```

```
Echo Reply (received/timeout) (14/0)
```

```
Total Time Elapsed 468 ms
```

The output of the **trace mpls multipath** command in the example shows the result of path discovery with the MPLS LSP Multipath Tree Trace feature as shown in [Figure 3](#).

Figure 3 MPLS LSP Multipath Tree Trace Path Discovery in a Sample Network



170603

Using DSCP to Request a Specific Class of Service in an Echo Reply: Example

The following example shows how to use DSCP to request a specific CoS in an echo reply:

```
Router# trace mpls multipath ipv4 10.1.1.150/32 reply dscp 50
```

```
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 448 ms
```


Controlling How a Responding Router Replies to an MPLS Echo Request: Example

The following example shows how to control how a responding router replies to an MPLS echo request:

```
Router# trace mpls multipath ipv4 10.1.1.150/32 reply mode router-alert
```

```
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 708 ms
```

Specifying the Output Interface for Echo Packets Leaving a Router for MPLS LSP Multipath Tree Trace: Example

The following example shows how to specify the output interface for echo packets leaving a router for the MPLS LSP Multipath Tree Trace feature:

```
Router# trace mpls multipath ipv4 10.1.1.150/32 output interface ethernet0/0
```

```
Tracing MPLS Label Switched Path to 10.1.1.150/32, 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 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.
```

```
0 10.1.111.101 MRU 1500 [Labels: 33 Exp: 0]
```

```
L
```

```
1 10.1.111.111 MRU 1500 [Labels: 33 Exp: 0] 40 ms
```

```
L
```

```

2 10.2.120.120 MRU 1500 [Labels: 33 Exp: 0] 20 ms
L
3 10.3.131.131 MRU 1500 [Labels: 34 Exp: 0] 20 ms
L
4 10.4.141.141 MRU 1504 [Labels: implicit-null Exp: 0] 20 ms !
5 10.5.150.150 16 ms

```

Setting the Pace of MPLS Echo Request Packet Transmission for MPLS LSP Multipath Tree Trace: Example

The following examples show how set the pace of MPLS echo request packet transmission for the MPLS LSP Multipath Tree Trace feature. The time between successive MPLS echo requests is set to 300 milliseconds in the first example and 400 milliseconds in the second example:

```
Router# trace mpls multipath ipv4 10.131.159.252/32 interval 300
```

```
Starting LSP Multipath Traceroute for 10.131.159.252/32
```

```
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.
```

```
LL!
```

```
Path 0 found,
output interface Et1/0 source 10.2.3.2 destination 127.0.0.0

```

```
Paths (found/broken/unexplored) (1/0/0)
```

```
Echo Request (sent/fail) (3/0)
```

```
Echo Reply (received/timeout) (3/0)
```

```
Total Time Elapsed 1604 ms

```

```
Router# trace mpls multipath ipv4 10.131.159.252/32 interval 400
```

```
Starting LSP Multipath Traceroute for 10.131.159.252/32
```

```
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.
```

```
LL!
```

```
Path 0 found,
output interface Et1/0 source 10.2.3.2 destination 127.0.0.0

```

```
Paths (found/broken/unexplored) (1/0/0)
```

```
Echo Request (sent/fail) (3/0)
```

```
Echo Reply (received/timeout) (3/0)
```

```
Total Time Elapsed 1856 ms

```

Notice that the elapsed time increases as you increase the interval size.

Enabling MPLS LSP Multipath Tree Trace to Detect LSP Breakages Caused by an Interface That Lacks an MPLS Configuration: Example

The following examples shows how to enable the MPLS LSP Multipath Tree Trace feature to detect LSP breakages caused by an interface that lacks an MPLS configuration:

```
Router# trace mpls multipath ipv4 10.1.1.150/32 force-explicit-null
```

```
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
```

This example shows the additional information provided when you add the **verbose** keyword to the command:

```
Router# trace mpls multipath ipv4 10.1.1.150/32 force-explicit-null verbose
```

```
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
```

```
0 10.1.111.101 10.1.111.111 MRU 1500 [Labels: 33/explicit-null Exp: 0/0] multipaths 0
```

```
L
```

```
1 10.1.111.111 10.2.121.121 MRU 1500 [Labels: 34/explicit-null Exp: 0/0] ret code 8
```

```
multipaths 2
```

```
L
```

```

    2 10.2.121.121 10.3.132.132 MRU 1500 [Labels: 34/explicit-null Exp: 0/0] ret code 8
multipaths 1
L
    3 10.3.132.132 10.4.140.240 MRU 1500 [Labels: 32/explicit-null Exp: 0/0] ret code 8
multipaths 1
L
    4 10.4.140.240 10.5.150.50 MRU 1504 [Labels: explicit-null Exp: 0] ret code 8 multipaths
1 !
    5 10.5.150.50, ret code 3 multipaths 0
LLL!
Path 1 found,
output interface Et0/0 source 10.1.111.101 destination 127.0.0.1
    0 10.1.111.101 10.1.111.111 MRU 1500 [Labels: 33/explicit-null Exp: 0/0] multipaths 0
L
    1 10.1.111.111 10.2.120.120 MRU 1500 [Labels: 33/explicit-null Exp: 0/0] ret code 8
multipaths 2
L
    2 10.2.120.120 10.3.131.131 MRU 1500 [Labels: 33/explicit-null Exp: 0/0] ret code 8
multipaths 2
L
    3 10.3.131.131 10.4.141.141 MRU 1500 [Labels: 34/explicit-null Exp: 0/0] ret code 8
multipaths 2
L
    4 10.4.141.141 10.5.150.150 MRU 1504 [Labels: explicit-null Exp: 0] ret code 8
multipaths 1
!
    5 10.5.150.150, ret code 3 multipaths 0
L!
Path 2 found,
output interface Et0/0 source 10.1.111.101 destination 127.0.0.5
    0 10.1.111.101 10.1.111.111 MRU 1500 [Labels: 33/explicit-null Exp: 0/0] multipaths 0
L
    1 10.1.111.111 10.2.120.120 MRU 1500 [Labels: 33/explicit-null Exp: 0/0] ret code 8
multipaths 2
L
    2 10.2.120.120 10.3.131.131 MRU 1500 [Labels: 33/explicit-null Exp: 0/0] ret code 8
multipaths 2
L
    3 10.3.131.131 10.4.140.140 MRU 1500 [Labels: 32/explicit-null Exp: 0/0] ret code 8
multipaths 2
L
    4 10.4.140.140 10.5.150.50 MRU 1504 [Labels: explicit-null Exp: 0] ret code 8 multipaths
1 !
    5 10.5.150.50, ret code 3 multipaths 0
LL!
Path 3 found,
output interface Et0/0 source 10.1.111.101 destination 127.0.0.7
    0 10.1.111.101 10.1.111.111 MRU 1500 [Labels: 33/explicit-null Exp: 0/0] multipaths 0
L
    1 10.1.111.111 10.2.120.120 MRU 1500 [Labels: 33/explicit-null Exp: 0/0] ret code 8
multipaths 2
L
    2 10.2.120.120 10.3.130.130 MRU 1500 [Labels: 34/explicit-null Exp: 0/0] ret code 8
multipaths 2
L
    3 10.3.130.130 10.4.140.40 MRU 1500 [Labels: 32/explicit-null Exp: 0/0] ret code 8
multipaths 1
L
    4 10.4.140.40 10.5.150.50 MRU 1504 [Labels: explicit-null Exp: 0] ret code 8 multipaths
1
!
    5 10.5.150.50, ret code 3 multipaths 0

Paths (found/broken/unexplored) (4/0/0)
Echo Request (sent/fail) (14/0)

```

```
Echo Reply (received/timeout) (14/0)
Total Time Elapsed 492 ms
```

Requesting That a Transit Router Validate the Target FEC Stack for MPLS LSP Multipath Tree Trace: Example

The following example shows how to request that a transit router validate the target FEC stack for the MPLS LSP Multipath Tree Trace feature:

```
Router# trace mpls multipath ipv4 10.1.1.150/32 flags fec ttl 5

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 464 ms
```

Target FEC stack validation is always done at the egress router when the **flags fec** keywords are specified in the **trace mpls multipath** command.

Setting the Number of Timeout Attempts for MPLS LSP Multipath Tree Trace: Example

The following example sets the number of timeout attempts for the MPLS LSP Multipath Tree Trace feature to four:

```
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 output shows a **trace mpls multipath** command that found one unexplored path, one successful path, and one broken path:

```
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.

```
LLL...
Path 0 Unexplorable,
  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 B
Path 2 Broken,
  output interface Et0/0 source 10.1.111.101 destination 127.0.0.7

Paths (found/broken/unexplored) (1/1/1)
Echo Request (sent/fail) (12/0)
Echo Reply (received/timeout) (8/4)
Total Time Elapsed 7868 ms
```

Additional References

The following sections provide references related to the MPLS EM—MPLS LSP Multipath Tree Trace feature.

Related Documents

Related Topic	Document Title
Concepts and configuration tasks for MPLS LSP ping or traceroute	MPLS Embedded Management—LSP Ping/Traceroute and AToM VCCV
Concepts and configuration for MPLS and other MPLS applications	Cisco IOS Multiprotocol Label Switching Configuration Guide
MPLS commands	Cisco IOS Multiprotocol Label Switching Command Reference
Troubleshooting procedures for MPLS	Cisco—MPLS Troubleshooting

Standards

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

MIBs

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

RFCs

RFC	Title
RFC 2113	<i>IP Router Alert Option</i>
RFC 3443	<i>Time To Live (TTL) Processing in Multi-Protocol Label Switching (MPLS) Networks</i>
RFC 4377	<i>Operations and Management (OAM) Requirements for Multi-Protocol Label Switched (MPLS) Networks</i>
RFC 4378	<i>A Framework for Multi-Protocol Label Switching (MPLS) Operations and Management (OAM)</i>
RFC 4379	<i>Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures</i>

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/techsupport</p>

Feature Information for MPLS EM—MPLS LSP Multipath Tree Trace

Table 3 lists the release history for this feature.

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, see the command reference documentation.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.



Note

Table 3 lists only the Cisco IOS software release that introduced support for a given feature in a given Cisco IOS software release train. Unless noted otherwise, subsequent releases of that Cisco IOS software release train also support that feature.

Table 3 Feature Information for MPLS EM—MPLS LSP Multipath Tree Trace

Feature Name	Releases	Feature Information
MPLS EM—MPLS LSP Multipath Tree Trace	12.2(31)SB2 12.2(33)SRB 12.4(20)T 12.2(33)SXI	<p>The MPLS EM—MPLS LSP Multipath Tree Trace feature provides the means to discover all the possible paths of a label switched path (LSP) between an egress and ingress router. Once discovered, these paths can be retested on a periodic basis using Multiprotocol Label Switching (MPLS) LSP ping or traceroute. This feature is an extension to the MPLS LSP traceroute functionality for the tracing of IPv4 LSPs.</p> <p>Cisco IOS MPLS Embedded Management (EM) is a set of standards and value-added services that facilitate the deployment, operation, administration, and management of MPLS-based networks in line with the fault, configuration, accounting, performance, and security (FCAPS) model.</p> <p>In Cisco IOS Release 12.2(31)SB2, this feature was introduced.</p> <p>In Cisco IOS Release 12.2(33)SRB, support was added for a Cisco IOS 12.2SR release.</p> <p>In Cisco IOS Release 12.(20)T, support was added for a Cisco IOS 12.4T release.</p> <p>In Cisco IOS Release 12.2(33)SXI, support was added for a Cisco IOS 12.2SX release.</p>

Table 3 Feature Information for MPLS EM—MPLS LSP Multipath Tree Trace (continued)

Feature Name	Releases	Feature Information
		<p>The following sections provide information about this feature:</p> <ul style="list-style-type: none"> • Overview of MPLS LSP Multipath Tree Trace, page 3 • Discovery of IPv4 Load Balancing Paths by MPLS LSP Multipath Tree Trace, page 3 • Echo Reply Return Codes Sent by the Router Processing Multipath LSP Tree Trace, page 4 Customizing the Default Behavior of MPLS Echo Packets, page 5 • Configuring MPLS LSP Multipath Tree Trace, page 7 • Discovering IPv4 Load Balancing Paths Using MPLS LSP Multipath Tree Trace, page 9 Monitoring LSP Paths Discovered by MPLS LSP Multipath Tree Trace Using MPLS LSP Traceroute, page 10 • Using DSCP to Request a Specific Class of Service in an Echo Reply, page 13 • Controlling How a Responding Router Replies to an MPLS Echo Request, page 14 • Specifying the Output Interface for Echo Packets Leaving a Router for MPLS LSP Multipath Tree Trace, page 16 • Setting the Pace of MPLS Echo Request Packet Transmission for MPLS LSP Multipath Tree Trace, page 17 • Enabling MPLS LSP Multipath Tree Trace to Detect LSP Breakages Caused by an Interface That Lacks an MPLS Configuration, page 18 • Requesting That a Transit Router Validate the Target FEC Stack for MPLS LSP Multipath Tree Trace, page 19 • Setting the Number of Timeout Attempts for MPLS LSP Multipath Tree Trace, page 21 <p>The following commands were introduced or modified: debug mpls lspv, echo, mpls oam, trace mpls, trace mpls multipath.</p>

Glossary

ECMP—equal-cost multipath. Multiple routing paths of equal cost that may be used for packet forwarding.

FEC—Forwarding Equivalence Class. A set of packets that can be handled equivalently for forwarding purposes and are thus suitable for binding to a single label. Examples include the set of packets destined for one address prefix and the packets in any flow.

flow—A set of packets traveling between a pair of hosts, or between a pair of transport protocol ports on a pair of hosts. For example, packets with the same source address, source port, destination address, and destination port might be considered a flow.

A flow is also a stream of data traveling between two endpoints across a network (for example, from one LAN station to another). Multiple flows can be transmitted on a single circuit.

localhost—A name that represents the host router (device). The localhost uses the reserved loopback IP address 127.0.0.1.

LSP—label switched path. A connection between two routers in which MPLS forwards the packets.

LSPV—Label Switched Path Verification. An LSP ping subprocess. It encodes and decodes MPLS echo requests and replies, and it interfaces with IP, MPLS, and AToM switching for sending and receiving MPLS echo requests and replies. At the MPLS echo request originator router, LSPV maintains a database of outstanding echo requests for which echo responses have not been received.

MPLS router alert label—An MPLS label of 1. An MPLS packet with a router alert label is redirected by the router to the Route Processor (RP) processing level for handling. This allows these packets to bypass any forwarding failures in hardware routing tables.

OAM—Operation, Administration, and Management.

punt—Redirect packets with a router alert from the line card or interface to Route Processor (RP) level processing for handling.

RP—Route Processor. The processor module in a Cisco 7000 series router that contains the CPU, system software, and most of the memory components that are used in the router. It is sometimes called a supervisory processor.

TTL—time-to-live. A parameter you can set that indicates the maximum number of hops a packet should take to reach its destination.

TLV—type, length, values. A block of information included in a Cisco Discovery Protocol address.

UDP—User Datagram Protocol. Connectionless transport layer protocol in the TCP/IP protocol stack. UDP is a simple protocol that exchanges datagrams without acknowledgments or guaranteed delivery, so error processing and retransmission must be handled by other protocols. UDP is defined in RFC 768.

XDR—eXternal Data Representation. Standard for machine-independent data structures developed by Sun Microsystems. Used to transport messages between the Route Processor (RP) and the line card.

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at www.cisco.com/go/trademarks. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1005R)

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

© 2006–2009 Cisco Systems, Inc. All rights reserved.