Network Working Group
Internet Draft
Category: Standards Track
Expiration Date: November 2007

George Swallow
Cisco Systems, Inc.

Kireeti Kompella
Juniper Networks, Inc.

Dan Tappan

May 2007

Label Switching Router Self-Test

draft-ietf-mpls-lsr-self-test-07.txt

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/1id-abstracts.html

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

Abstract

This document defines a means for a Label-Switching Router to verify that its data plane is functioning for certain key Multi-Protocol Label Switching applications, including unicast forwarding and traffic engineering tunnels. A new Loopback FEC

type is defined to allow an upstream neighbor to assist in the testing at very low cost. MPLS Verification Request and MPLS Verification Reply messages are defined to do the actual probing.

Contents

1. Introduction

This document defines extentions to RFC 4379 [LSP-PING] (which is generally known as Label-Switched-Path (LSP) Ping) to provide a means for a Label-Switching Router (LSR) to verify that its data plane is functioning for certain key Multi-Protocol Label Switching (MPLS) applications, including unicast forwarding based on the Label Distribution Protocol (LDP) [LDP] and traffic engineering tunnels based on [RSVP-TE]. MPLS Verification Request and MPLS Verification Reply messages are defined to do the actual probing. The pings are sent to an upstream neighbor, looped back through the LSR under test and intercepted, by means of time-to-live (TTL) expiration by a downstream neighbor.

In order to minimize the load on upstream LSRs a loopback FEC Type is defined. Labels advertised with this FEC Type are referred to as loopback labels. Receipt of a packet labeled with a loopback label will cause the advertising LSR to pop the label off the label stack and send the packet out the advertised interface.

Use of a loopback mechnism allows an LSR to test label entries which are not currently in use. For example many LSRs advertise label mappings for all IPv4 routes to all of their neighbors. For some portion of these their neighbor LSR is not currently upstream and the label entry is not used. But if the neighbors best path to a destination changes, that route and the associated label entry will be used. An LSR can loop traffic through a "non-upstream" LSR because that LSR is acting only on the loopback label and not on the underlying label associated with the actual forwarding equivalence class (FEC) being tested. In this way label entries can be verified prior to the occurrence of a routing change.

Some routing protocols, most notably Open Shortest-Path-First (OSPF) [OSPF] have no means of exchanging the "Link Local Identifiers" used to identify unnumbered links and components of bundled links. These test procedures can be used to associate the neighbor's interfaces with the probing LSRs interfaces. This is achieved by simply having the TTL of the LSP Ping expire one hop sooner, i.e. at the testing LSR itself.

1.1. Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [KEYWORDS].

2. Loopback FEC

The Loopback FEC type is defined to enable an upstream neighbor to assist in LSR self-testing at very low cost. This FEC causes the loopback to occur in the data plane without control plane involvement beyond the initial LDP exchange and data-plane setup. The FEC also carries information to indicate the desired encapsulation should it be the only label in a received label stack. Values are defined for IPv4 and IPv6.

An LSR uses a Loopback FEC to selectively advertise loopback labels to its neighbor LSRs. Each loopback label is bound to a particular interface. For multi-access links, a unique label for each neighbor is required, since the link-level address is derived from the label lookup. When an MPLS packet with its top label set to a loopback label is received from an interface over which that label was advertised, the loopback label is popped and the packet is sent on the interface to which the loopback label was bound. If the label-stack only contains the one loopback label, the encapsulation of the packet is determined by the FEC Type.

TTL treatment for loopback labels follows the Uniform model. I.e. the TTL carried in the loopback label is decremented and copied to the exposed label or IP header as the case may be.

2.1. Loopback FEC Element

FEC element type 130 is used.   The FEC element is encoded as follows: (note: 130 is provisionally assigned, the actual value will be assigned by IANA.)

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     130       |      Res      | If & Prot Type|   Id Length   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Interface Identifier                      |
|                              "                                |
|                              "                                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Reserved (Res)

MUST be set to zero on transmission and ignored on receipt.

Interface & Protocol Type

#	Type	Interface Identifier
1	IPv4 Numbered	IPv4 Address
2	IPv4 Unnumbered	A 32 bit Link Identifier as defined in [RFC3477]
3	IPv6 Numbered	IPv6 Address
4	IPv6 Unnumbered	A 32 bit Link Identifier as defined in [RFC3477]

Note that these type values also indicate the encapsulation (IPv4 or IPv6) for payloads that have a label stack containing only a loopback label.

Identifier Length

Length of the interface identifier in octets. The length is 4 bytes for the unnumbered types and IPv4, 16 bytes for IPv6.

Address

An identifier encoded according to the Identifier Type field.

2.2. LDP Procedures

It is RECOMMENDED that loopback labels only be distributed in response to a Label Request message, irrespective of the label advertisement mode of the LDP session. However it is recognized that in certain cases such as OSPF with unnumbered links, the upstream LSR may not have sufficiently detailed information of the neighbor's link identifier to form the request. In these cases, the downstream LSR MAY be configured to make unsolicited advertisements.

3. Data Plane Self Test

A self test operation involves three LSRs, the LSR doing the test, an upstream neighbor and a downstream LSR. Upstream here is with respect to the flow of the test (which in some cases could be different than the normal sense of upstream in IP routing). We refer to these as LSRs T, U, and D respectively. In order to minimize the processing load on LSR-D, two new LSP Ping messages are defined, called the MPLS Data Plane Verification Request and the MPLS Data Plane Verification Reply. These messages are used to allow LSR-T to obtain the label stack, address and interface information of LSR-D.

The packet flow is shown below. Although the figure shows LSR-D adjacent to LSR-T it may in some cases be an arbitrary number of hops away.

```
+-------+       +-------+       +-------+
|     ,-|-------|<DPVRq |       |       |
|     `-|-------|-------|-------|->     |
|       |       |       |       |       |
|       |       |     <-|-------|<DPVRp |
+-------+       +-------+       +-------+
  LSR-U           LSR-T           LSR-D

DPVRq: MPLS Data Plane Verification Request
DPVRp: MPLS Data Plane Verification Reply
```

Figure 1: Self Test Message Flow

In order to perform a test on an incoming label stack, LSR-T forms an MPLS Data Plane Verification Request. LSR-T prepends the packet with the incoming label stack being tested and the loopback label received from LSR-U. The TTL values are set such that they will expire at LSR-D. LSR-T then forwards the packet to LSR-U.

LSR-U receives the packet and performs normal MPLS forwarding. That is, the loopback label is popped, the TTL is decremented and propagated (in this case) to the exposed label.

LSR-T receives the packet and performs normal MPLS forwarding. If everything is functioning as expected this will cause the packet to arrive at LSR-D with a TTL of 1.

In this example, we assume that all is working properly. The TTL expires at LSR-D causing it to receive the packet. LSR-D notes the the interface and the label stack on which the packet was received and records these in an Interface and Label Stack TLV. This Object is sent to LSR-T in an MPLS Data Plane Verification Reply message.

3.1. Data Plane Verification Request / Reply Messages

Two new LSP Ping messages are defined for LSR self test. The purpose of the new messages is three fold. First the timestamps are removed to minimize processing. Second the message type allows simple recognition that minimal processing is necessary to service this request. Third, the Verification Request message itself conveys the the request, thus a Verification Request message with no Objects is both legal and normal.

The definitions of all fields in the messages are identical to those found in [LSP-PING].

The new message types are: (Provisionally; to be assigned)

Type	Message
3	MPLS Data Plane Verification Request
4	MPLS Data Plane Verification Reply

The messages have the following format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Version Number        |         MUST Be Zero          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Message Type |   Reply mode  |  Return Code  | Return Subcode|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Sender's Handle                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Sequence Number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            TLVs ...                           |
.                                                               .
.                                                               .
.                                                               .
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The MPLS Data Plane Verification Request message MAY contain the following objects:

Type #	Object
3	Pad
10	Reply TOS Byte
11 (provisional)	IPv4 Reply-to Address
12 (provisional)	IPv6 Reply-to Address
64512-65535	Vendor Private TLVs

The MPLS Data Plane Verification Reply message MAY contain the following objects:

Type #	Object
7	Interface and Label Stack
9	Errored TLVs
64512-65535	Vendor Private TLVs

3.2. UDP Port

MPLS Data Plane Verification Request messages MAY be sent to port 3503 as is used for [LSP-PING]. However to aid implementations that wish to handle these messages at a lower level than MPLS Echo Request messages another UDP port, <tbd>, is provided. Port <tbd> SHOULD be used by default. The source UDP port, as in [LSP-PING] is chosen by the sender.

3.3. Reply-To Address Object

In order to perform detailed diagnostics of a particular failing flow in the face of ECMP, it is useful to be able to use the exact source and destination addresses of that flow. The Reply-To Object is an optional TLV in a MPLS Data Plane Verification Request message. The Object has two formats, type 11 for IPv4 and type 12 for IPv6 (to be assigned by IANA).

3.3.1. IPv4 Reply-To Address Object

The length of an IPv4 Reply-to Address object is 4 octets; the value field has the following format:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                     Reply-to IPv4 Address                     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Reply-to IPv4 Address

The address to which the MPLS Data Plane Verification Reply message is to be sent.

3.3.2. IPv6 Reply-To Address Object

The length of an IPv6 Reply-to Address object is 16 octets; the value field has the following format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Reply-to IPv6 Address                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 Reply-to IPv6 Address (Cont.)                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 Reply-to IPv6 Address (Cont.)                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 Reply-to IPv6 Address (Cont.)                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Reply-to IPv6 Address

The address to which the MPLS Data Plane Verification Reply message is to be sent.

3.4. Sending procedures

In order to perform a test on an incoming labeled or unlabeled packet, an LSR first determines the expected outgoing label stack, next hop router and next hop interface.

The LSR creates an MPLS Data Plane Verification Request message.

In normal use, the source address is set to an address belonging to the LSR and the destination set to an address in the range of 127/8. The incoming label stack (if any) is prepended to the packet. The TTL of these labels and the packet header SHOULD be set to appropriate values - 2 for those labels and/or header which will be processed by this node when the packet is looped back; 1 for those labels and/or header which will be carried through. Finally the loopback label bound to the incoming interface is prepended to the packet. In the case of an otherwise unlabeled packet the label's FEC MUST indicate the appropriate IP version. The TTL is set such that it will have the value of 3 on the wire.

The packet is sent to the upstream neighbor on an interface for which the loopback label is valid.

In diagnostic situations, the source and destination addresses MAY be set to any value. In this case, a Reply-to IPv4 or IPv6 Address object MUST be included. The IP TTL MUST be set to 1. The TTL of labels other than the loopback label MUST be set to appropriate values - 2 for those labels which will be process by this LSR when the packet is looped back; 1 for those labels which will be carried

through.

In some MPLS deployments TTL hiding is used to make a providers network appear as a single hop. That is the TTL in the imposed label does not reflect the TTL of the received packet. It is RECOMMENDED that testing of label imposition SHOULD NOT be performed in such circumstances as the Verification Request will in most case travel multiple hops.

3.5. Receiving procedures

An LSR X that receives an MPLS Verification Request message formats a MPLS Verification Reply message. The Sender's Handle and Sequence Number are copied from the Request message.

X then parses the packet to ensure that it is a well-formed packet, and that the TLVs that are not marked "Ignore" are understood. If not, X SHOULD set the Return Code set to "Malformed echo request received" or "TLV not understood" (as appropriate), and the Subcode set to zero. In the latter case, the misunderstood TLVs (only) are included in the reply.

If the Verification Request is good, X MUST note the interface and label stack of the received Verification Request and format this information as a Downstream Verification object. This object is included in the MPLS Verification Reply message. The Return Code and Subcode MUST be set to zero, indicating "No return code".

The source address of the Reply message MUST be an address of the replying LSR. If the request included a Reply-to IPv4 or IPv6 Address object, the MPLS Data Plane Verification Reply message MUST be sent to that address. Otherwise the Reply message is sent to the source address of the Verification Request message.

An LSR MUST be capable of filtering addresses that are to be replied to. If a filter has been invoked (i.e. configured) and an address does not pass the filter, then a reply MUST NOT be sent, and the event SHOULD be logged.

3.6. Upstream Neighbor Verification

To verify that an upstream neighbor is properly echoing packets an LSR may send an MPLS Data Plane Verification Request packet with the TTL set so that the packet will expire upon reaching reaching itself. This procedure not only tests that the neighbor is correctly processing the loopback label, it also allows the node to verify the neighbor's interface mapping.

```
                    +-------+       +-------+
                    |       |       |       |
                    |     ,-|-------|<DPVRq |
                    |     `-|-------|->     |
                    |       |       |       |
                    +-------+       +-------+
                      LSR-U           LSR-T

          DPVRq: MPLS Data Plane Verification Request
```

Figure 2: Upstream Neighbor Verification

No TLVs need to be included in the MPLS Data Plane Verification Request. By noting the Sender's Handle and Sequence Number, as well as the loopback label, LSR-T is able to detect that a) the packet was looped, and b) determine (or verify) the interface on which the packet was received.

4. Security Considerations

Were loopback labels widely known, they might be subject to abuse. It is therefore RECOMMENDED that loopback labels only be shared between trusted neighbors. Further, if the loopback labels are drawn from a per-platform label space, or any other label space shared across multiple LDP sessions, it is RECOMMENDED that all loopback labels be filtered from a session except those labels pertaining to interfaces directly connected to the neighbor participating in that session.

5. IANA Considerations

This document makes the following codepoint assigments (pending IANA action):

Registry	Codepoint	Purpose
UDP Port	tbd	MPLS Verification Request
LSP Ping Message Type	3	MPLS Data Plane Verification Request
LSP Ping Message Type	4	MPLS Data Plane Verification Reply
LSP Ping Object Type	11	IPv4 Reply-to Address
LSP Ping Object Type	12	IPv6 Reply-to Address

6. Acknowledgments

The authors would like to thank Vanson Lim, Tom Nadeau, and Bob Thomas for their comments and suggestions.

7. References

7.1. Normative References

[LDP] Andersson, L. et al., "LDP Specification", RFC 3036, January 2001.

[LSP-PING] Kompella, K. and G. Swallow, "Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures", RFC 4379, February 2006.

[RFC3477] Kompella, K. and Y. Rekhter, "Signalling Unnumbered Links in Resource ReSerVation Protocol - Traffic Engineering (RSVP-TE)", January 2003.

[KEYWORDS] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

7.2. Informative References

[RSVP-TE] Awduche, D., et al, "RSVP-TE: Extensions to RSVP for LSP tunnels", RFC 3209, December 2001.

[OSPF]   Moy, J., "OSPF Version 2", STD 54, RFC 2328, April 1998.

8. Authors' Addresses

Kireeti Kompella
Juniper Networks, Inc.
1194 N. Mathilda Ave.
Sunnyvale, CA 94089
Email: kireeti@juniper.net

George Swallow
Cisco Systems, Inc.
1414 Massachusetts Ave
Boxborough, MA 01719

Email: swallow@cisco.com

Dan Tappan
Boxborough, MA 01719

Email: dtappan@alum.mit.edu

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at

http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Full Copyright Notice

Copyright (C) The IETF Trust (2007). This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.