

MPLS Working Group  
Internet-Draft  
Intended status: Informational  
Expires: April 17, 2015

R. Torvi  
R. Bonica  
Juniper Networks  
M. Conn  
D. Pacella  
L. Tomotaki  
M. Wygant  
Verizon  
October 14, 2014

**LSP Self-Ping**  
**draft-bonica-mpls-self-ping-00**

**Abstract**

This memo describes LSP Self-ping. An ingress LSR can use LSP Self-ping to verify that an LSP is ready to carry traffic.

**Status of This Memo**

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on April 17, 2015.

**Copyright Notice**

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in [Section 4.e](#) of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

<a href="#">1.</a>	Introduction . . . . .	<a href="#">2</a>
<a href="#">2.</a>	LSP Self Ping Procedures . . . . .	<a href="#">3</a>
<a href="#">3.</a>	Rejected Approaches . . . . .	<a href="#">5</a>
<a href="#">4.</a>	IANA Considerations . . . . .	<a href="#">6</a>
<a href="#">5.</a>	Security Considerations . . . . .	<a href="#">6</a>
<a href="#">6.</a>	Acknowledgements . . . . .	<a href="#">6</a>
<a href="#">7.</a>	Normative References . . . . .	<a href="#">6</a>
	Authors' Addresses . . . . .	<a href="#">7</a>

## [1.](#) Introduction

An ingress Label Switching Router (LSR) can use RSVP-TE [[RFC3209](#)] to establish an MPLS Label Switched Path [[RFC3032](#)]. The following paragraphs provide an overview of RSVP-TE procedures.

The ingress LSR calculates an explicit path between itself and an egress LSR. It then formats an RSVP PATH message, including an Explicit Route Object (ERO). The ERO represents the explicit path between the ingress and egress LSRs.

The ingress LSR forwards the PATH message in the direction of the egress LSR, following the path defined by the ERO. Each transit LSR that receives the PATH message executes admission control procedures. If the transit LSR admits the LSP, it reserves bandwidth (if necessary) and sends the PATH message downstream, to the next node in the ERO.

When the egress LSR receives the PATH message, it allocates a label. The label can be implicit null, explicit null or non-null. The egress LSR then installs forwarding state (if necessary), and constructs an RSVP RESV message. The RESV message includes a Label Object containing the allocated label.

The egress LSR sends the RESV message upstream towards the ingress LSR. The RESV message visits the same transit LSRs that the PATH message visited, but in reverse order. Each transit LSR allocates a label, updates its forwarding state and updates the RESV message. As a result, the RESV message contains a Label Object and the Label Object contains the locally allocated label. Next, the transit LSR sends the RESV message upstream, along the explicit path.



The ingress LSR receives the RESV message and installs forwarding state. Once the ingress LSR installs forwarding state it can forward traffic through the LSP.

An implementation can optimize the procedure described above by allowing LSRs to send a RESV messages upstream before installing forwarding state. This optimization is desirable, because it allows LSRs to install forwarding state in parallel, thus accelerating the process of LSP signaling and setup. However, this optimization creates a race condition. When the ingress LSR receives a RESV message, some downstream LSRs may have not yet completed the process of forwarding state installation. If the ingress sends traffic over the LSP, the traffic will be black-holed until forwarding state has been installed on all downstream LSRs.

The ingress LSP can prevent back-holing by verifying the LSPs readiness to carry traffic before forwarding traffic through it. This memo describes LSP Self-ping. Ingress LSRs can use LSP Self-Ping to verify that an LSP is ready to carry traffic.

## **2. LSP Self Ping Procedures**

In order to verify that an LSP is ready to carry traffic, the ingress LSR creates a short-lived LSP Self-ping session. All session state is maintained locally on the ingress LSR. Session state includes the following:

- o Session-id: A 16-bit number that identifies the session
- o verification-status: A boolean variable indicating whether LSP readiness has been verified. The initial value of this variable is FALSE.
- o retries: The number of times that the ingress LSR probes the LSP before giving up. The initial value of this variable is determined by configuration.
- o retry-timer: The number of milliseconds that the LSR waits after sending a probe. The initial value of this variable is determined by configuration.

The ingress LSR executes the following procedure until verification-status equals TRUE or retries is less than 1:

- o Format a probe message
- o Send the probe message through the LSP under test



- o Set a timer to expire in retry-timer milliseconds
- o Wait until either a) a probe message associated with the session returns or b) the timer expires. If a probe associated with the session returns, set verification-status to TRUE. Otherwise, decrement retries. Optionally, increase the value of retry-timer according to an appropriate back off algorithm.

If the protocol messages used to establish the LSP were delivered over IPv4 [[RFC0791](#)] the probe message is an ICMPv4 [[RFC0792](#)] Echo Reply. If the protocol messages used to establish the LSP were delivered over IPv6 [[RFC2460](#)] the probe message is an ICMPv6 [[RFC4443](#)] Echo Reply. In either case, the contents of the ICMP message are as follows:

- o Source Address equals the address of the egress LSR
- o Destination Address equals the address of the ingress LSR
- o Time to Live (TTL) / Hop Count equals 255
- o DSCP equals CS6 (0x48) [[RFC4594](#)]
- o ICMP Identifier equals Session-id
- o ICMP Sequence equals retries

If the LSP under test is ready to carry traffic, the egress LSR receives the Echo Reply message. The Echo Reply message can arrive at the egress LSR with or without an MPLS header, depending on whether the LSP under test executes penultimate hop-popping procedures. If the Echo Reply message arrives at the egress LSR with an MPLS header, the egress LSR removes that header.

The egress LSR forwards the Echo Reply message to its destination, the ingress LSR. The egress LSR forwards the Echo Reply message exactly as it would forward any other packet. If the egress LSR's most preferred route to the ingress LSR is through an LSP, the egress LSR forwards the Echo Reply message through that LSP. However, if the egress LSR's most preferred route to the ingress LSR is not through an LSP, the egress LSR forwards the Echo Reply message without MPLS encapsulation.

If the ingress LSR receives an ICMP Echo Reply message with ICMP identifier equal to the Session-ID, it sets the verification-status to TRUE. The ICMP Sequence number does not have to match the last Sequence Number sent.



LSP Self-ping fails if ICMP Echo Reply messages cannot be delivered from the egress LSR to the ingress LSR. For example, LSP Self-ping will fail if the intervening network blocks ICMP Echo Reply messages.

When an LSP Self-ping session terminates, it returns the value of verification-status to the invoking protocol. For example, assume that RSVP-TE invokes LSP Self-ping as part of the LSP set-up procedure. If LSP Self-ping returns TRUE, RSVP-TE makes the LSP under test available for forwarding. However, if LSP Self-ping returns FALSE, RSVP-TE takes appropriate remedial actions.

### **3. Rejected Approaches**

In a rejected approach, the ingress LSR uses LSP-Ping [[RFC4379](#)] to verify LSP readiness to carry traffic. This approach was rejected for the following reasons.

While an ingress LSR can control its control plane overhead due to LSP Ping, an egress LSR has no such control. This is because each ingress LSR can, on its own, control the rate of the LSP Ping originated by the LSR, while an egress LSR must respond to all the LSP Pings originated by various ingresses. Furthermore, when an MPLS Echo Request reaches an egress LSR it is sent to the control plane of the egress LSR, which makes egress LSR processing overhead of LSP Ping well above the overhead of its data plane (MPLS/IP forwarding). These factors make LSP Ping problematic as a tool for detecting LSP readiness to carry traffic when dealing with a large number of LSPs.

By contrast, LSP Self-ping does not consume any control plane resources at the egress LSR, and relies solely on the data plane of the egress LSR, making it more suitable as a tool for checking LSP readiness when dealing with a large number of LSPs.

In another rejected approach, the ingress LSR does not verify LSP readiness. Alternatively, it sets a timer when it receives an RSVP RESV message and does not forward traffic through the LSP until the timer expires. This approach was rejected because it is impossible to determine the optimal setting for this timer. If the timer value is set too low, it does not prevent black-holing. If the timer value is set too high, it slows down the process of LSP signalling and setup.

Moreover, the above-mentioned timer is configured on a per-router basis. However, its optimum value is determined by a network-wide behavior. Therefore, changes in the network could require changes to the value of the timer, making the optimal setting of this timer a moving target.





#### **4. IANA Considerations**

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

#### **5. Security Considerations**

ICMP messages are easily forged. Therefore, an attacker can send the ingress LSR a forged ICMP Echo Reply message, causing the ingress LSR to terminate the LSP Self-ping session prematurely.

#### **6. Acknowledgements**

Thanks to Yakov Rekhter for his contributions to this document.

#### **7. Normative References**

- [RFC0791] Postel, J., "Internet Protocol", STD 5, [RFC 791](#), September 1981.
- [RFC0792] Postel, J., "Internet Control Message Protocol", STD 5, [RFC 792](#), September 1981.
- [RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", [RFC 2460](#), December 1998.
- [RFC3032] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y., Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack Encoding", [RFC 3032](#), January 2001.
- [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", [RFC 3209](#), December 2001.
- [RFC4379] Kompella, K. and G. Swallow, "Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures", [RFC 4379](#), February 2006.
- [RFC4443] Conta, A., Deering, S., and M. Gupta, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", [RFC 4443](#), March 2006.
- [RFC4594] Babiarz, J., Chan, K., and F. Baker, "Configuration Guidelines for DiffServ Service Classes", [RFC 4594](#), August 2006.



Authors' Addresses

Ravi Torvi  
Juniper Networks

Email: rtorvi@juniper.net

Ron Bonica  
Juniper Networks

Email: rbonica@juniper.net

Michael Conn  
Verizon

Email: michael.e.conn@verizon.com

Dante Pacella  
Verizon

Email: dante.j.pacella@verizon.com

Luis Tomotaki  
Verizon

Email: luis.tomotaki@verizon.com

Mark Wygant  
Verizon

Email: mark.wygant@verizon.com

