

Network Work group
Internet-Draft
Intended status: Standards Track
Expires: August 3, 2016

N. Kumar
G. Swallow
C. Pignataro
Cisco Systems, Inc.
N. Akiya
Big Switch Networks
S. Kini
Individual
H. Gredler
Juniper Networks
M. Chen
Huawei
January 31, 2016

Label Switched Path (LSP) Ping/Trace for Segment Routing Networks Using
MPLS Dataplane
[draft-kumarkini-mpls-spring-lsp-ping-05](#)

Abstract

Segment Routing architecture leverages the source routing and tunneling paradigms and can be directly applied to MPLS data plane. A node steers a packet through a controlled set of instructions called segments, by prepending the packet with a Segment Routing header.

The segment assignment and forwarding semantic nature of Segment Routing raises additional consideration for connectivity verification and fault isolation in LSP with Segment Routing architecture. This document illustrates the problem and describe a mechanism to perform LSP Ping and Traceroute on Segment Routing network over MPLS data plane.

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 August 3, 2016.

Copyright Notice

Copyright (c) 2016 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

- [1.](#) Introduction [3](#)
- [2.](#) Requirements notation [3](#)
- [3.](#) Terminology [3](#)
- [4.](#) Challenges with Existing mechanism [4](#)
 - [4.1.](#) Path validation in Segment Routing networks [4](#)
 - [4.2.](#) Service Label [5](#)
- [5.](#) Segment ID sub-TLV [5](#)
 - [5.1.](#) IPv4 Prefix Node Segment ID [5](#)
 - [5.2.](#) IPv6 Prefix Node Segment ID [6](#)
 - [5.3.](#) IGP Adjacency Segment ID [7](#)
- [6.](#) Extension to Downstream Mapping TLV [8](#)
- [7.](#) Procedures [9](#)
 - [7.1.](#) FECs in Target FEC Stack TLV [9](#)
 - [7.2.](#) FEC Stack Change sub-TLV [10](#)
 - [7.3.](#) Segment ID POP Operation [10](#)
 - [7.4.](#) Segment ID Check [10](#)
 - [7.5.](#) TTL Consideration for traceroute [12](#)
- [8.](#) Issues with non-forwarding labels [12](#)
- [9.](#) IANA Considerations [13](#)
 - [9.1.](#) New Target FEC Stack Sub-TLVs [13](#)
- [10.](#) Security Considerations [13](#)
- [11.](#) Acknowledgement [13](#)
- [12.](#) Contributing Authors [13](#)
- [13.](#) References [14](#)
 - [13.1.](#) Normative References [14](#)
 - [13.2.](#) Informative References [15](#)
- Authors' Addresses [15](#)

1. Introduction

[I-D.ietf-spring-segment-routing] introduces and explains Segment Routing architecture that leverages the source routing and tunneling paradigms. A node steers a packet through a controlled set of instructions called segments, by prepending the packet with Segment Routing header. A detailed definition about Segment Routing architecture is available in [I-D.ietf-spring-segment-routing] and different use-cases are discussed in [I-D.filsfils-spring-segment-routing-use-cases]

The Segment Routing architecture can be directly applied to MPLS data plane in a way that, the Segment identifier (Segment ID) will be of 20-bits size and Segment Routing header is the label stack.

Multi Protocol Label Switching (MPLS) has defined in [RFC4379] a simple and efficient mechanism to detect data plane failures in Label Switched Paths (LSP) by specifying information to be carried in an MPLS "echo request" and "echo reply" for the purposes of fault detection and isolation, and mechanisms for reliably sending the echo reply. The functionality is modeled after the ping/traceroute paradigm (ICMP echo request [RFC0792]) and is typically referred to as LSP ping and LSP traceroute.

Unlike LDP or RSVP which are the other well-known MPLS control plane protocols, segment assignment in Segment Routing architecture is not hop-by-hop basis.

This nature of Segment Routing raises additional consideration for fault detection and isolation in Segment Routing network. This document illustrates the problem and describe a mechanism to perform LSP Ping and Traceroute on Segment Routing network over MPLS data plane.

2. Requirements notation

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 [RFC2119].

3. Terminology

This document uses the terminologies defined in [I-D.ietf-spring-segment-routing], [RFC4379], and so the readers are expected to be familiar with the same.

4. Challenges with Existing mechanism

This document defines sub-TLVs for the Target FEC Stack TLV and explains how they can be used to tackle below challenges.

4.1. Path validation in Segment Routing networks

[RFC4379] defines the OAM machinery that helps with fault detection and isolation in MPLS dataplane path with the use of various Target FEC Stack Sub-TLV that are carried in MPLS Echo Request packets and used by the responder for FEC validation. While it is obvious that new Sub-TLVs need to be assigned, the unique nature of Segment Routing architecture raises a need for additional machinery for path validation. This section discuss the challenges as below:

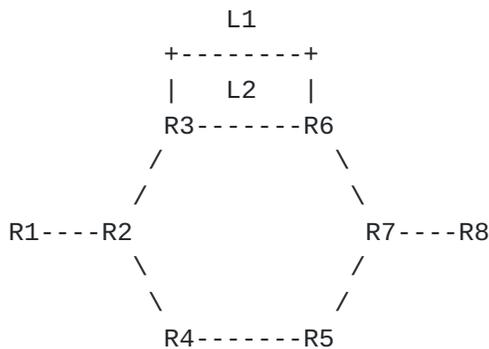


Figure 1: Segment Routing network

The Node segment IDs for R1, R2, R3, R4, R5, R6, R7 and R8 are 5001, 5002, 5003, 5004, 5005, 5006, 5007, 5008 respectively.

- 9136 --> Adjacency Segment ID from R3 to R6 over link L1.
- 9236 --> Adjacency Segment ID from R3 to R6 over link L2.
- 9124 --> Adjacency segment ID from R2 to R4.
- 9123 --> Adjacency Segment ID from R2 to R3.

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

Assume in above topology, R1 sends traffic with segment stack as {9124, 5008} so that the path taken will be R1-R2-R4-R5-R7-R8. If the Adjacency Segment ID 9124 is misprogrammed in R2 to send the

packet to R1 or R3, it will still be delivered to R8 but is not via the expected path.

MPLS traceroute may help with detecting such deviation in above mentioned scenario. However, in a different example, it may not be helpful. For example if R3, due to misprogramming, forwards packet with Adjacency Segment ID 9236 via link L1 while it is expected to be forwarded over Link L2.

4.2. Service Label

A Segment ID can represent a service based instruction. An Segment Routing header can have label stack entries where the label represents a service to be applied along the path. Since these labels are part of the label stack, they can influence the path taken by a packet and consequently have implications on MPLS OAM. In [section 6.5](#) of this document, it is described how the procedures of [RFC4379] can be applied to in the absence of service-labels in [Section 6.5](#). Additional considerations for service labels are included in [Section 7](#) and requires further discussion.

5. Segment ID sub-TLV

The format of the following Segment ID sub-TLVs follows the philosophy of Target FEC Stack TLV carrying FECs corresponding to each label in the label stack. When operated with the procedures defined in [RFC4379], this allows LSP ping/traceroute operations to function when Target FEC Stack TLV contains more FECs than received label stack at responder nodes.

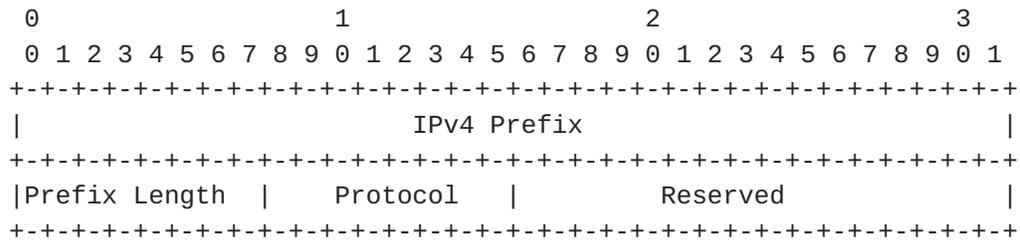
Three new sub-TLVs are defined for TLVs type 1, 16 and 21.

sub-Type	Value Field
-----	-----
TBD1	IPv4 Prefix Node Segment ID
TBD2	IPv6 Prefix Node Segment ID
TBD3	Adjacency Segment ID

Service Segments and FRR will be considered in future version.

5.1. IPv4 Prefix Node Segment ID

The format is as below:



IPv4 Prefix

This field carries the IPv4 prefix to which the Node Segment ID is assigned. If the prefix is shorter than 32 bits, trailing bits SHOULD be set to zero.

Prefix Length

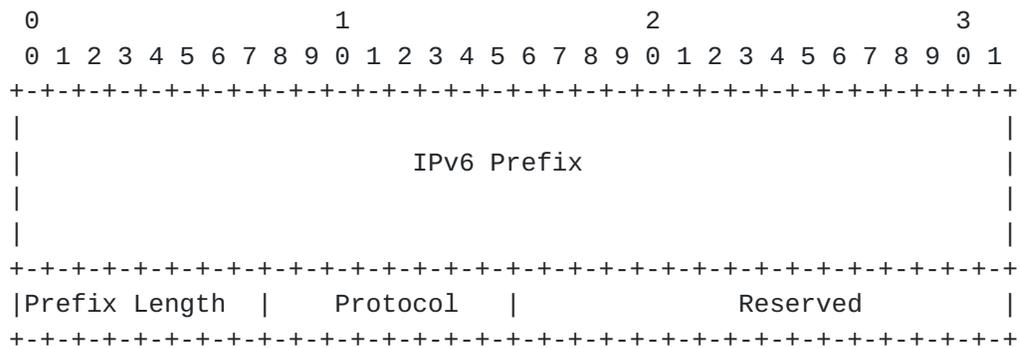
The Prefix Length field is one octet, it gives the length of the prefix in bits (values can be 1 - 32).

Protocol

Set to 1 if the IGP protocol is OSPF and 2 if IGP protocol is ISIS.

5.2. IPv6 Prefix Node Segment ID

The format is as below:



IPv6 Prefix

This field carries the IPv6 prefix to which the Node Segment ID is assigned. If the prefix is shorter than 128 bits, trailing bits SHOULD be set to zero.

Prefix Length

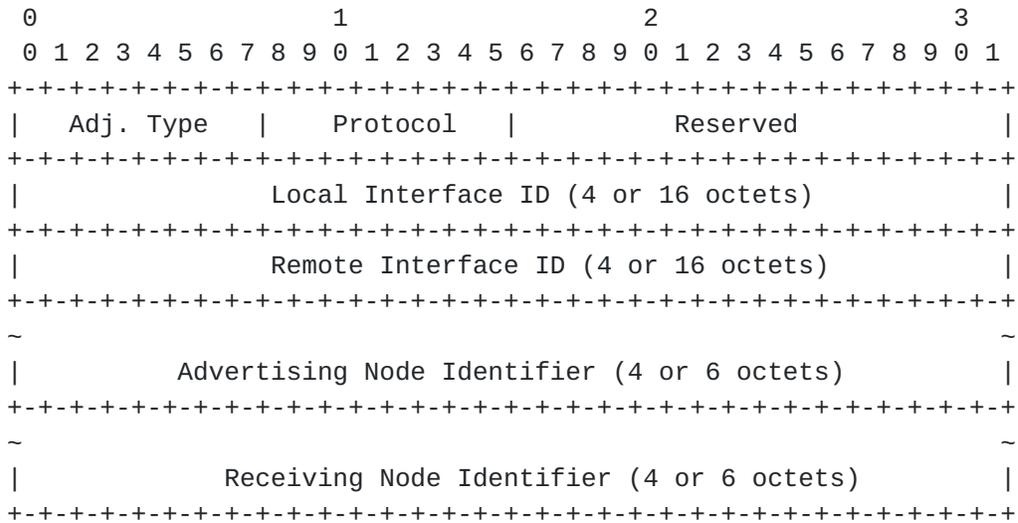
The Prefix Length field is one octet, it gives the length of the prefix in bits (values can be 1 - 128).

Protocol

Set to 1 if the IGP protocol is OSPF and 2 if IGP protocol is ISIS.

5.3. IGP Adjacency Segment ID

The format is as below:



Adj. Type

Set to 1, when the Adjacency Segment is Parallel Adjacency as defined in section 3.5.1 of [I-D.ietf-spring-segment-routing]. Set to 4, when the Adjacency segment is IPv4 based and is not a parallel adjacency. Set to 6, when the Adjacency segment is IPv6 based and is not a parallel adjacency.

Protocol

Set to 1 if the IGP protocol is OSPF and 2 if IGP protocol is ISIS

Local Interface ID

An identifier that is assigned by local LSR for a link on which Adjacency Segment ID is bound. This field is set to local link address (IPv4 or IPv6). In case of unnumbered, 32 bit link identifier defined in [\[RFC4203\]](#), [\[RFC5307\]](#) is used. If the Adjacency Segment ID represents parallel adjacencies (Section 3.5.1 of [\[I-D.ietf-spring-segment-routing\]](#)) this field MUST be set to zero.

Remote Interface ID

An identifier that is assigned by remote LSR for a link on which Adjacency Segment ID is bound. This field is set to remote (downstream neighbor) link address (IPv4 or IPv6). In case of unnumbered, 32 bit link identifier defined in [\[RFC4203\]](#), [\[RFC5307\]](#) is used. If the Adjacency Segment ID represents parallel adjacencies (Section 3.5.1 of [\[I-D.ietf-spring-segment-routing\]](#)) this field MUST be set to zero.

Advertising Node Identifier

Specifies the advertising node identifier. When Protocol is set to 1, then the 32 rightmost bits represent OSPF Router ID and if protocol is set to 2, this field carries 48 bit ISIS System ID.

Receiving Node Identifier

Specifies the downstream node identifier. When Protocol is set to 1, then the 32 rightmost bits represent OSPF Router ID and if protocol is set to 2, this field carries 48 bit ISIS System ID.

6. Extension to Downstream Mapping TLV

In an echo reply, the Downstream Mapping TLV [\[RFC4379\]](#) is used to report for each interface over which a FEC could be forwarded. For a FEC, there are multiple protocols that may be used to distribute label mapping. The "Protocol" field of the Downstream Mapping TLV is used to return the protocol that is used to distribute a specific a label. The following protocols are defined in [section 3.2 of \[RFC4379\]](#):

Protocol #	Signaling Protocol
-----	-----
0	Unknown
1	Static
2	BGP
3	LDP
4	RSVP-TE

With segment routing, OSPF or ISIS can be used for label distribution, this document adds two new protocols as follows:

Protocol #	Signaling Protocol
-----	-----
5	OSPF
6	ISIS

7. Procedures

This section describes aspects of LSP Ping and traceroute operations that require further considerations beyond [\[RFC4379\]](#).

7.1. FECs in Target FEC Stack TLV

When LSP echo request packets are generated by an initiator, FECs carried in Target FEC Stack TLV may need to have deviating contents. This document outlines expected Target FEC Stack TLV construction mechanics by initiator for known scenarios.

Ping

Initiator MUST include FEC(s) corresponding to the destination segment.

Initiator MAY include FECs corresponding to some or all of segments imposed in the label stack by the initiator to communicate the segments traversed.

Traceroute

Initiator MUST initially include FECs corresponding to all of segments imposed in the label stack.

When a received echo reply contains FEC Stack Change TLV with one or more of original segment(s) being popped, initiator MAY remove corresponding FEC(s) from Target FEC Stack TLV in the next (TTL+1) traceroute request.

When a received echo reply does not contain FEC Stack Change TLV, initiator MUST NOT attempt to remove FEC(s) from Target FEC Stack TLV in the next (TTL+1) traceroute request. Note that Downstream Label field of DSMAP/DDMAP contains hints on how initiator may be able to update the contents of next Target FEC Stack TLV. However, such hints are ambiguous without full understanding of PHP capabilities.

7.2. FEC Stack Change sub-TLV

[Section 3.3.1.3 of \[RFC6424\]](#) defines a new sub-TLV that a router must include when the FEC stack changes.

The network node which advertised the Node Segment ID is responsible for generating FEC Stack Change sub-TLV of &pop& operation for Node Segment ID, regardless of if PHP is enabled or not.

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

7.3. Segment ID POP Operation

The forwarding semantic of Node Segment ID with PHP flag is equivalent to usage of implicit Null in MPLS protocols. Adjacency Segment ID is also similar in a sense that it can be thought as next hop destined locally allocated segment that has PHP enabled. Procedures described in [Section 4.4 of \[RFC4379\]](#) relies on Stack-D and Stack-R explicitly having Implicit Null value. It may simplify implementations to reuse Implicit Null for Node Segment ID PHP and Adjacency Segment ID cases. However, it is technically incorrect for Implicit Null value to externally appear. Therefore, implicit Null MUST NOT be placed in Stack-D and Interface and Label Stack TLV for Node Segment ID PHP and Adjacency Segment ID cases.

7.4. Segment ID Check

If the Label-stack-depth is 0 and Target FEC Stack Sub-TLV at FEC-stack-depth is TBD1 (IPv4 Prefix Node Segment ID), the responder should set Best return code to 10 if any below conditions fail: /* The responder LSR is to check if it is the egress of the IPv4 Prefix Node Segment ID described in the Target FEC Stack Sub-TLV, and if the FEC was advertised with the PHP bit set.*/

- * Validate that Node Segment ID is advertised for IPv4 Prefix.
- * Validate that Node Segment ID is advertisement of PHP bit.

If the Label-stack-depth is more than 0 and Target FEC Stack Sub-TLV at FEC-stack-depth is TBD1 (IPv4 Prefix Node Segment ID), the responder is to set Best return code to 10 if any below conditions fail:

- * Validate that Node Segment ID is advertised for IPv4 Prefix.

If the Label-stack-depth is 0 and Target FEC Sub-TLV at FEC-stack-depth is TBD2 (IPv6 Prefix Node Segment ID), set Best return code to 10 if any below conditions fail: /* The LSR needs to check if its being a tail-end for the LSP and have the prefix advertised with PHP bit set*/

- * Validate that Node Segment ID is advertised for IPv6 Prefix.
- * Validate that Node Segment ID is advertised of PHP bit.

If the Label-stack-depth is 0 and Target FEC Sub-TLV at FEC-stack-depth is TBD2 (IPv6 Prefix Node Segment ID), set Best return code to 10 if any below conditions fail:

- * Validate that Node Segment ID is advertised for IPv6 Prefix.

If the Label-stack-depth is 0 and Target FEC sub-TLV at FEC-stack-depth is TBD3 (Adjacency Segment ID), set Best return code to (error code TBD) if any below conditions fail:

When the Adj.Type is 1 (Parallel Adjacency):

- + Validate that Receiving Node Identifier is local IGP identifier.
- + Validate that Adjacency Segment ID is advertised by Advertising Node Identifier of Protocol in local IGP database.

When the Adj.Type is 4 or 6:

- + Validate that Remote Interface ID matches the local identifier of the interface (Interface-I) on which the packet was received.
- + Validate that Receiving Node Identifier is local IGP identifier.
- + Validate that IGP Adjacency Segment ID is advertised by Advertising Node Identifier of Protocol in local IGP database.

7.5. TTL Consideration for traceroute

LSP Traceroute operation can properly traverse every hop of Segment Routing network in Uniform Model described in [RFC3443]. If one or more LSRs employ Short Pipe Model described in [RFC3443], then LSP Traceroute may not be able to properly traverse every hop of Segment Routing network due to absence of TTL copy operation when outer label is popped. In such scenario, following TTL manipulation technique MAY be used.

When tracing a LSP according to the procedures in [RFC4379] the TTL is incremented by one in order to trace the path sequentially along the LSP. However when a source routed LSP has to be traced there are as many TTLs as there are labels in the stack. The LSR that initiates the traceroute SHOULD start by setting the TTL to 1 for the tunnel in the LSP's label stack it wants to start the tracing from, the TTL of all outer labels in the stack to the max value, and the TTL of all the inner labels in the stack to zero. Thus a typical start to the traceroute would have a TTL of 1 for the outermost label and all the inner labels would have TTL 0. If the FEC Stack TLV is included it should contain only those for the inner stacked tunnels. The lack of an echo response or the Return Code/Subcode should be used to diagnose the tunnel as described in [RFC4379]. When the tracing of a tunnel in the stack is complete, then the next tunnel in the stack should be traced. The end of a tunnel can be detected from the "Return Code" when it indicates that the responding LSR is an egress for the stack at depth 1. Thus the traceroute procedures in [RFC4379] can be recursively applied to traceroute a source routed LSP.

8. Issues with non-forwarding labels

Source stacking can be optionally used to apply services on the packet at a LSR along the path, where a label in the stack is used to trigger service application. A data plane failure detection and isolation mechanism should provide its functionality without applying these services. This is mandatory for services that are stateful, though for stateless services [RFC4379] could be used as-is. It MAY also provide a mechanism to detect and isolate faults within the service function itself.

To prevent services from being applied to an "echo request" packet, the TTL of service labels MUST be 0. However TTL processing rules of a service label must be the same as any MPLS label. Due to this a TTL of 0 in the service label would prevent the packet from being forwarded beyond the LSR that provides the service. To avoid this problem, the originator of the "echo request" MUST NOT include the service label in the label stack of an echo request above the tunnel

label of the tunnel that is being currently traced. In other words the ingress must remove all service-labels above the label of the tunnel being currently traced, but retain service labels below it when sending the echo request. Note that load balancing may affect the path when the service labels are removed, resulting in a newer path being traversed. However this new path is potentially different only up to the LSR that provides the service. Since this portion of the path was traced when the tunnels above this tunnel in the stack were traced and followed the exact path as the source routed LSP, this should not be a major concern. Sometimes the newer path may have a problem that was not in the original path resulting in a false positive. In such a case the original path can be traversed by changing the label stack to reach the intermediate LSR with labels that route along each hop explicitly.

9. IANA Considerations

9.1. New Target FEC Stack Sub-TLVs

IANA is requested to assign 3 new Sub-TLVs from "Sub-TLVs for TLV Types 1, 16 and 21" sub-registry.

Sub-Type	Sub-TLV Name	Reference
TBD1	IPv4 Prefix Node Segment ID	Section 4.1 (this document)
TBD2	IPv6 Prefix Node Segment ID	Section 4.2 (this document)
TBD3	Adjacency Segment ID	Section 4.3 (this document)

10. Security Considerations

This document defines additional Sub-TLVs and follows the mechanism defined in [[RFC4379](#)]. So all the security consideration defined in [[RFC4379](#)] will be applicable for this document and in addition it does not impose any security challenges to be considered.

11. Acknowledgement

The authors would like to thank Stefano Previdi, Les Ginsberg, Balaji Rajagopalan and Harish Sitaraman for their review and comments.

The authors would like to thank Loa Andersson for his comments and recommendation to merge drafts.

12. Contributing Authors

Tarek Saad
 Cisco Systems
 Email: tsaad@cisco.com

Siva Sivabalan
Cisco Systems
Email: msiva@cisco.com

Balaji Rajagopalan
Juniper Networks
Email: balajir@juniper.net

13. References

13.1. Normative References

- [I-D.filsfils-spring-segment-routing-use-cases]
Filsfils, C., Francois, P., Previdi, S., Decraene, B., Litkowski, S., Horneffer, M., Milojevic, I., Shakir, R., Ytti, S., Henderickx, W., Tantsura, J., Kini, S., and E. Crabbe, "Segment Routing Use Cases", [draft-filsfils-spring-segment-routing-use-cases-01](#) (work in progress), October 2014.
- [I-D.ietf-spring-segment-routing]
Filsfils, C., Previdi, S., Decraene, B., Litkowski, S., and r. rjs@rob.sh, "Segment Routing Architecture", [draft-ietf-spring-segment-routing-07](#) (work in progress), December 2015.
- [RFC0792] Postel, J., "Internet Control Message Protocol", STD 5, [RFC 792](#), DOI 10.17487/RFC0792, September 1981, <<http://www.rfc-editor.org/info/rfc792>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC3443] Agarwal, P. and B. Akyol, "Time To Live (TTL) Processing in Multi-Protocol Label Switching (MPLS) Networks", [RFC 3443](#), DOI 10.17487/RFC3443, January 2003, <<http://www.rfc-editor.org/info/rfc3443>>.
- [RFC4203] Kompella, K., Ed. and Y. Rekhter, Ed., "OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", [RFC 4203](#), DOI 10.17487/RFC4203, October 2005, <<http://www.rfc-editor.org/info/rfc4203>>.

- [RFC4379] Kompella, K. and G. Swallow, "Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures", [RFC 4379](#), DOI 10.17487/RFC4379, February 2006, <<http://www.rfc-editor.org/info/rfc4379>>.
- [RFC5307] Kompella, K., Ed. and Y. Rekhter, Ed., "IS-IS Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", [RFC 5307](#), DOI 10.17487/RFC5307, October 2008, <<http://www.rfc-editor.org/info/rfc5307>>.
- [RFC6424] Bahadur, N., Kompella, K., and G. Swallow, "Mechanism for Performing Label Switched Path Ping (LSP Ping) over MPLS Tunnels", [RFC 6424](#), DOI 10.17487/RFC6424, November 2011, <<http://www.rfc-editor.org/info/rfc6424>>.

13.2. Informative References

- [RFC6291] Andersson, L., van Helvoort, H., Bonica, R., Romascanu, D., and S. Mansfield, "Guidelines for the Use of the "OAM" Acronym in the IETF", [BCP 161](#), [RFC 6291](#), DOI 10.17487/RFC6291, June 2011, <<http://www.rfc-editor.org/info/rfc6291>>.
- [RFC6425] Saxena, S., Ed., Swallow, G., Ali, Z., Farrel, A., Yasukawa, S., and T. Nadeau, "Detecting Data-Plane Failures in Point-to-Multipoint MPLS - Extensions to LSP Ping", [RFC 6425](#), DOI 10.17487/RFC6425, November 2011, <<http://www.rfc-editor.org/info/rfc6425>>.

Authors' Addresses

Nagendra Kumar
Cisco Systems, Inc.
7200 Kit Creek Road
Research Triangle Park, NC 27709
US

Email: naikumar@cisco.com

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

Email: swallow@cisco.com

Carlos Pignataro
Cisco Systems, Inc.
7200 Kit Creek Road
Research Triangle Park, NC 27709-4987
US

Email: cpignata@cisco.com

Nobo Akiya
Big Switch Networks

Email: nobo.akiya.dev@gmail.com

Sriganesh Kini
Individual

Email: sriganeshkini@gmail.com

Hannes Gredler
Juniper Networks

Email: hannes@juniper.net

Mach(Guoyi) Chen
Huawei

Email: mach.chen@huawei.com

