

Internet Engineering Task Force (IETF)  
Request for Comments: 9259  
Category: Standards Track  
ISSN: 2070-1721

Z. Ali  
C. Filsfils  
Cisco Systems  
S. Matsushima  
Softbank  
D. Voyer  
Bell Canada  
M. Chen  
Huawei  
June 2022

## Operations, Administration, and Maintenance (OAM) in Segment Routing over IPv6 (SRv6)

### Abstract

This document describes how the existing IPv6 mechanisms for ping and traceroute can be used in a Segment Routing over IPv6 (SRv6) network. The document also specifies the OAM flag (O-flag) in the Segment Routing Header (SRH) for performing controllable and predictable flow sampling from segment endpoints. In addition, the document describes how a centralized monitoring system performs a path continuity check between any nodes within an SRv6 domain.

### Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at <https://www.rfc-editor.org/info/rfc9259>.

### Copyright Notice

Copyright (c) 2022 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 (<https://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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

### Table of Contents

1.	Introduction
1.1.	Requirements Language
1.2.	Abbreviations
1.3.	Terminology and Reference Topology
2.	OAM Mechanisms
2.1.	OAM Flag in the Segment Routing Header
2.1.1.	OAM Flag Processing
2.2.	OAM Operations
3.	Security Considerations
4.	Privacy Considerations
5.	IANA Considerations
6.	References
6.1.	Normative References
6.2.	Informative References
Appendix A.	Illustrations
A.1.	Ping in SRv6 Networks
A.1.1.	Pinging an IPv6 Address via a Segment List
A.1.2.	Pinging a SID
A.2.	Traceroute in SRv6 Networks
A.2.1.	Traceroute to an IPv6 Address via a Segment List
A.2.2.	Traceroute to a SID
A.3.	Hybrid OAM Using the OAM Flag
A.4.	Monitoring of SRv6 Paths
	Acknowledgements
	Contributors
	Authors' Addresses

## 1. Introduction

As Segment Routing over IPv6 (SRv6) [RFC8402] simply adds a new type of Routing Extension Header, existing IPv6 OAM mechanisms can be used in an SRv6 network. This document describes how the existing IPv6 mechanisms for ping and traceroute can be used in an SRv6 network. This includes illustrations of pinging an SRv6 Segment Identifier (SID) to verify that the SID is reachable and is locally programmed at the target node. This also includes illustrations for tracerouting to an SRv6 SID for hop-by-hop fault localization as well as path tracing to a SID.

This document also introduces enhancements for the OAM mechanism for SRv6 networks that allow controllable and predictable flow sampling from segment endpoints using, e.g., the IP Flow Information Export (IPFIX) protocol [RFC7011]. Specifically, the document specifies the OAM flag (O-flag) in the SRH as a marking bit in the user packets to trigger telemetry data collection and export at the segment endpoints.

This document also outlines how the centralized OAM technique in [RFC8403] can be extended for SRv6 to perform a path continuity check between any nodes within an SRv6 domain. Specifically, the document illustrates how a centralized monitoring system can monitor arbitrary SRv6 paths by creating loopback probes that originate and terminate at the centralized monitoring system.

### 1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

## 1.2. Abbreviations

The following abbreviations are used in this document:

SID: Segment Identifier

SL: Segments Left

SR: Segment Routing

SRH: Segment Routing Header [RFC8754]

SRv6: Segment Routing over IPv6 [RFC8402]

PSP: Penultimate Segment Pop [RFC8986]

USP: Ultimate Segment Pop [RFC8986]

ICMPv6: Internet Control Message Protocol for the Internet Protocol version 6 [RFC4443]

IS-IS: Intermediate System to Intermediate System

OSPF: Open Shortest Path First [RFC2328]

IGP: Interior Gateway Protocol (e.g., OSPF and IS-IS)

BGP-LS: Border Gateway Protocol - Link State [RFC8571]

## 1.3. Terminology and Reference Topology

The terminology and simple topology in this section are used for illustration throughout the document.

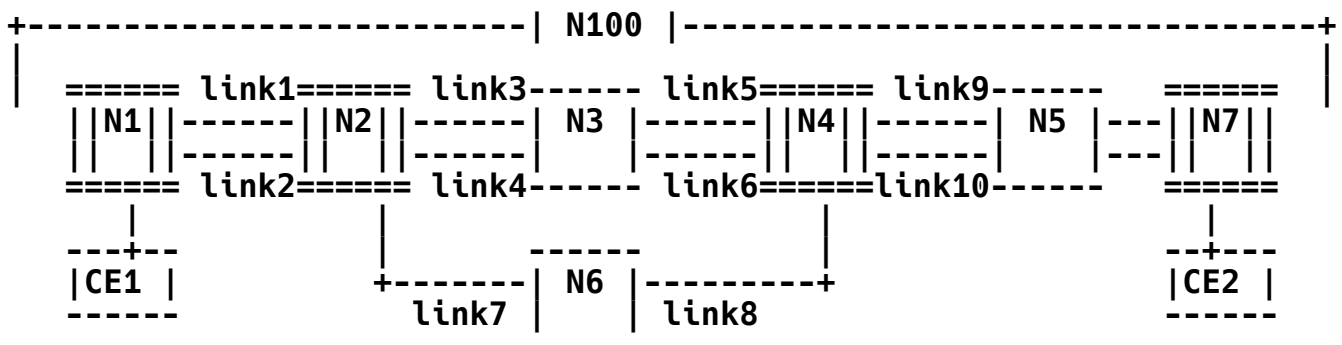


Figure 1: Reference Topology

In the reference topology:

- \* Node j has an IPv6 loopback address 2001:db8:L:j::/128.
- \* Nodes N1, N2, N4, and N7 are SRv6-capable nodes.
- \* Nodes N3, N5, and N6 are IPv6 nodes that are not SRv6-capable nodes. Such nodes are referred to as "non-SRv6-capable nodes".
- \* CE1 and CE2 are Customer Edge devices of any data plane capability (e.g., IPv4, IPv6, and L2).
- \* A SID at node j with locator block 2001:db8:K::/48 and function U is represented by 2001:db8:K:j:U::.
- \* Node N100 is a controller.
- \* The IPv6 address of the nth link between nodes i and j at the i side is represented as 2001:db8:i:j:in::. For example, in Figure 1, the IPv6 address of link6 (the second link between nodes N3 and N4) at node N3 is 2001:db8:3:4:32::. Similarly, the IPv6 address of link5 (the first link between nodes N3 and N4) at node N3 is 2001:db8:3:4:31::.
- \* 2001:db8:K:j:Xin:: is explicitly allocated as the End.X SID at node j towards neighbor node i via the nth link between nodes i and j. For example, 2001:db8:K:2:X31:: represents End.X at node N2 towards node N3 via link3 (the first link between nodes N2 and N3). Similarly, 2001:db8:K:4:X52:: represents the End.X at node N4 towards node N5 via link10 (the second link between nodes N4 and N5). Please refer to [RFC8986] for a description of End.X SID.
- \* A SID list is represented as <S1, S2, S3>, where S1 is the first SID to visit, S2 is the second SID to visit, and S3 is the last SID to visit along the SR path.
- \* (SA,DA) (S3, S2, S1; SL)(payload) represents an IPv6 packet with:
  - IPv6 header with source address SA, destination address DA, and SRH as the next header
  - SRH with SID list <S1, S2, S3> with SegmentsLeft = SL

Note the difference between the < > and ( ) symbols: <S1, S2, S3> represents a SID list where S1 is the first SID and S3 is the last SID to traverse. (S3, S2, S1; SL) represents the same SID list but encoded in the SRH format where the rightmost SID in the SRH is the first SID and the leftmost SID in the SRH is the last SID. When referring to an SR Policy in a high-level use case, it is simpler to use the <S1, S2, S3> notation. When referring to an illustration of the detailed packet behavior, the (S3, S2, S1; SL) notation is more convenient.

  - (payload) represents the payload of the packet.

## 2. OAM Mechanisms

This section defines OAM enhancements for SRv6 networks.

## 2.1. OAM Flag in the Segment Routing Header

[RFC8754] describes the Segment Routing Header (SRH) and how SR-capable nodes use it. The SRH contains an 8-bit Flags field.

This document defines the following bit in the SRH Flags field to carry the 0-flag:

```

  0 1 2 3 4 5 6 7
+--+--+--+--+--+--+
|      |0|          |
+--+--+--+--+--+--+
```

Where:

0-flag: OAM flag in the SRH Flags field defined in [RFC8754].

### 2.1.1. OAM Flag Processing

The 0-flag in the SRH is used as a marking bit in user packets to trigger telemetry data collection and export at the segment endpoints.

An SR domain ingress edge node encapsulates packets traversing the SR domain as defined in [RFC8754]. The SR domain ingress edge node MAY use the 0-flag in the SRH for marking the packet to trigger the telemetry data collection and export at the segment endpoints. Based on local configuration, the SR domain ingress edge node may implement a classification and sampling mechanism to mark a packet with the 0-flag in the SRH. Specification of the classification and sampling method is outside the scope of this document.

This document does not specify the data elements that need to be exported and the associated configurations. Similarly, this document does not define any formats for exporting the data elements. Nonetheless, without the loss of generality, this document assumes that the IP Flow Information Export (IPFIX) protocol [RFC7011] is used for exporting the traffic flow information from the network devices to a controller for monitoring and analytics. Similarly, without the loss of generality, this document assumes that requested information elements are configured by the management plane through data set templates (e.g., as in IPFIX [RFC7012]).

Implementation of the 0-flag is OPTIONAL. If a node does not support the 0-flag, then it simply ignores it upon reception. If a node supports the 0-flag, it can optionally advertise its potential via control plane protocol(s).

The following is appended to line S01 of the pseudocode associated with the SID S (as defined in Section 4.3.1.1 of [RFC8754]) when N receives a packet destined to S, S is a local SID, and the 0-flag is processed.

S01.1. IF the 0-flag is set and local configuration permits

```
0-flag processing {  
  a. Make a copy of the packet.  
  b. Send the copied packet, along with a timestamp,  
    to the OAM process for telemetry data collection  
    and export.      ;; Ref1  
}
```

Ref1: To provide an accurate timestamp, an implementation should copy and record the timestamp as soon as possible during packet processing. Timestamp and any other metadata are not carried in the packet forwarded to the next hop.

Please note that the 0-flag processing happens before execution of regular processing of the local SID S. Specifically, line S01.1 of the pseudocode specified in this document is inserted between lines S01 and S02 of the pseudocode defined in Section 4.3.1.1 of [RFC8754].

Based on the requested information elements configured by the management plane through data set templates [RFC7012], the OAM process exports the requested information elements. The information elements include parts of the packet header and/or parts of the packet payload for flow identification. The OAM process uses information elements defined in IPFIX [RFC7011] and Packet Sampling (PSAMP) [RFC5476] for exporting the requested sections of the mirrored packets.

If the penultimate segment of a segment list is a PSP SID, telemetry data from the ultimate segment cannot be requested. This is because, when the penultimate segment is a PSP SID, the SRH is removed at the penultimate segment, and the 0-flag is not processed at the ultimate segment.

The processing node **MUST** rate-limit the number of packets punted to the OAM process to a configurable rate. This is to avoid impacting the performance of the OAM and telemetry collection processes. Failure to implement the rate limit can lead to a denial-of-service attack, as detailed in Section 3.

The OAM process **MUST NOT** process the copy of the packet or respond to any Upper-Layer header (like ICMP, UDP, etc.) payload to prevent multiple evaluations of the datagram.

The OAM process is expected to be located on the routing node processing the packet. Although the specification of the OAM process or the external controller operations are beyond the scope of this document, the OAM process **SHOULD NOT** be topologically distant from the routing node, as this is likely to create significant security and congestion issues. How to correlate the data collected from different nodes at an external controller is also outside the scope of this document. Appendix A illustrates use of the 0-flag for implementing a hybrid OAM mechanism, where the "hybrid" classification is based on [RFC7799].

## 2.2. OAM Operations

IPv6 OAM operations can be performed for any SRv6 SID whose behavior

allows Upper-Layer header processing for an applicable OAM payload (e.g., ICMP, UDP).

Ping to an SRv6 SID is used to verify that the SID is reachable and is locally programmed at the target node. Traceroute to a SID is used for hop-by-hop fault localization as well as path tracing to a SID. Appendix A illustrates the ICMPv6-based ping and UDP-based traceroute mechanisms for ping and traceroute to an SRv6 SID. Although this document only illustrates ICMPv6-based ping and UDP-based traceroute to an SRv6 SID, the procedures are equally applicable to other OAM mechanisms that probe an SRv6 SID (e.g., Bidirectional Forwarding Detection (BFD) [RFC5880], Seamless BFD (S-BFD) [RFC7880], and Simple Two-way Active Measurement Protocol (STAMP) probe message processing [STAMP-SR]). Specifically, as long as local configuration allows the Upper-Layer header processing of the applicable OAM payload for SRv6 SIDs, the existing IPv6 OAM techniques can be used to target a probe to a (remote) SID.

IPv6 OAM operations can be performed with the target SID in the IPv6 destination address without an SRH or with an SRH where the target SID is the last segment. In general, OAM operations to a target SID may not exercise all of its processing depending on its behavior definition. For example, ping to an End.X SID [RFC8986] only validates the SID is locally programmed at the target node and does not validate switching to the correct outgoing interface. To exercise the behavior of a target SID, the OAM operation should construct the probe in a manner similar to a data packet that exercises the SID behavior, i.e. to include that SID as a transit SID in either an SRH or IPv6 DA of an outer IPv6 header or as appropriate based on the definition of the SID behavior.

### 3. Security Considerations

[RFC8754] defines the notion of an SR domain and use of the SRH within the SR domain. The use of OAM procedures described in this document is restricted to an SR domain. For example, similar to SID manipulation, 0-flag manipulation is not considered a threat within the SR domain. Procedures for securing an SR domain are defined in Sections 5.1 and 7 of [RFC8754].

As noted in Section 7.1 of [RFC8754], compromised nodes within the SR domain may mount attacks. The 0-flag may be set by an attacking node attempting a denial-of-service attack on the OAM process at the segment endpoint node. An implementation correctly implementing the rate limiting described in Section 2.1.1 is not susceptible to that denial-of-service attack. Additionally, SRH flags are protected by the Hashed Message Authentication Code (HMAC) TLV, as described in Section 2.1.2.1 of [RFC8754]. Once an HMAC is generated for a segment list with the 0-flag set, it can be used for an arbitrary amount of traffic using that segment list with the 0-flag set.

The security properties of the channel used to send exported packets marked by the 0-flag will depend on the specific OAM processes used. An on-path attacker able to observe this OAM channel could conduct traffic analysis, or potentially eavesdropping (depending on the OAM configuration), of this telemetry for the entire SR domain from such

a vantage point.

This document does not impose any additional security challenges to be considered beyond the security threats described in [RFC4884], [RFC4443], [RFC0792], [RFC8754], and [RFC8986].

#### 4. Privacy Considerations

The per-packet marking capabilities of the 0-flag provide a granular mechanism to collect telemetry. When this collection is deployed by an operator with the knowledge and consent of the users, it will enable a variety of diagnostics and monitoring to support the OAM and security operations use cases needed for resilient network operations. However, this collection mechanism will also provide an explicit protocol mechanism to operators for surveillance and pervasive monitoring use cases done contrary to the user's consent.

#### 5. IANA Considerations

IANA has registered the following in the "Segment Routing Header Flags" subregistry in the "Internet Protocol Version 6 (IPv6) Parameters" registry:

Bit	Description	Reference
2	0-flag	RFC 9259

Table 1

#### 6. References

##### 6.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8754] Filsfils, C., Ed., Dukes, D., Ed., Previdi, S., Leddy, J., Matsushima, S., and D. Voyer, "IPv6 Segment Routing Header (SRH)", RFC 8754, DOI 10.17487/RFC8754, March 2020, <<https://www.rfc-editor.org/info/rfc8754>>.
- [RFC8986] Filsfils, C., Ed., Camarillo, P., Ed., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "Segment Routing over IPv6 (SRv6) Network Programming", RFC 8986, DOI 10.17487/RFC8986, February 2021, <<https://www.rfc-editor.org/info/rfc8986>>.

##### 6.2. Informative References



- [RFC0792] Postel, J., "Internet Control Message Protocol", STD 5, RFC 792, DOI 10.17487/RFC0792, September 1981, <<https://www.rfc-editor.org/info/rfc792>>.
- [RFC2328] Moy, J., "OSPF Version 2", STD 54, RFC 2328, DOI 10.17487/RFC2328, April 1998, <<https://www.rfc-editor.org/info/rfc2328>>.
- [RFC4443] Conta, A., Deering, S., and M. Gupta, Ed., "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", STD 89, RFC 4443, DOI 10.17487/RFC4443, March 2006, <<https://www.rfc-editor.org/info/rfc4443>>.
- [RFC4884] Bonica, R., Gan, D., Tappan, D., and C. Pignataro, "Extended ICMP to Support Multi-Part Messages", RFC 4884, DOI 10.17487/RFC4884, April 2007, <<https://www.rfc-editor.org/info/rfc4884>>.
- [RFC5476] Claise, B., Ed., Johnson, A., and J. Quittek, "Packet Sampling (PSAMP) Protocol Specifications", RFC 5476, DOI 10.17487/RFC5476, March 2009, <<https://www.rfc-editor.org/info/rfc5476>>.
- [RFC5837] Atlas, A., Ed., Bonica, R., Ed., Pignataro, C., Ed., Shen, N., and JR. Rivers, "Extending ICMP for Interface and Next-Hop Identification", RFC 5837, DOI 10.17487/RFC5837, April 2010, <<https://www.rfc-editor.org/info/rfc5837>>.
- [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <<https://www.rfc-editor.org/info/rfc5880>>.
- [RFC7011] Claise, B., Ed., Trammell, B., Ed., and P. Aitken, "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information", STD 77, RFC 7011, DOI 10.17487/RFC7011, September 2013, <<https://www.rfc-editor.org/info/rfc7011>>.
- [RFC7012] Claise, B., Ed. and B. Trammell, Ed., "Information Model for IP Flow Information Export (IPFIX)", RFC 7012, DOI 10.17487/RFC7012, September 2013, <<https://www.rfc-editor.org/info/rfc7012>>.
- [RFC7799] Morton, A., "Active and Passive Metrics and Methods (with Hybrid Types In-Between)", RFC 7799, DOI 10.17487/RFC7799, May 2016, <<https://www.rfc-editor.org/info/rfc7799>>.
- [RFC7880] Pignataro, C., Ward, D., Akiya, N., Bhatia, M., and S. Pallagatti, "Seamless Bidirectional Forwarding Detection (S-BFD)", RFC 7880, DOI 10.17487/RFC7880, July 2016, <<https://www.rfc-editor.org/info/rfc7880>>.
- [RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment

Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <<https://www.rfc-editor.org/info/rfc8402>>.

- [RFC8403] Geib, R., Ed., Filsfils, C., Pignataro, C., Ed., and N. Kumar, "A Scalable and Topology-Aware MPLS Data-Plane Monitoring System", RFC 8403, DOI 10.17487/RFC8403, July 2018, <<https://www.rfc-editor.org/info/rfc8403>>.
- [RFC8571] Ginsberg, L., Ed., Previdi, S., Wu, Q., Tantsura, J., and C. Filsfils, "BGP - Link State (BGP-LS) Advertisement of IGP Traffic Engineering Performance Metric Extensions", RFC 8571, DOI 10.17487/RFC8571, March 2019, <<https://www.rfc-editor.org/info/rfc8571>>.
- [RFC9197] Brockners, F., Ed., Bhandari, S., Ed., and T. Mizrahi, Ed., "Data Fields for In Situ Operations, Administration, and Maintenance (IOAM)", RFC 9197, DOI 10.17487/RFC9197, May 2022, <<https://www.rfc-editor.org/info/rfc9197>>.
- [STAMP-SR] Gandhi, R., Ed., Filsfils, C., Voyer, D., Chen, M., Janssens, B., and R. Foote, "Performance Measurement Using Simple TWAMP (STAMP) for Segment Routing Networks", Work in Progress, Internet-Draft, draft-ietf-spring-stamp-srpm-03, 1 February 2022, <<https://datatracker.ietf.org/doc/html/draft-ietf-spring-stamp-srpm-03>>.

## Appendix A. Illustrations

This appendix shows how some of the existing IPv6 OAM mechanisms can be used in an SRv6 network. It also illustrates an OAM mechanism for performing controllable and predictable flow sampling from segment endpoints. How the centralized OAM technique in [RFC8403] can be extended for SRv6 is also described in this appendix.

### A.1. Ping in SRv6 Networks

The existing mechanism to perform the reachability checks, along the shortest path, continues to work without any modification. Any IPv6 node (SRv6-capable or non-SRv6-capable) can initiate, transit, and egress a ping packet.

The following subsections outline some additional use cases of ICMPv6 ping in SRv6 networks.

#### A.1.1. Pinging an IPv6 Address via a Segment List

If an SRv6-capable ingress node wants to ping an IPv6 address via an arbitrary segment list <S1, S2, S3>, it needs to initiate an ICMPv6 ping with an SR header containing the SID list <S1, S2, S3>. This is illustrated using the topology in Figure 1. The user issues a ping from node N1 to a loopback of node N5 via segment list <2001:db8:K:2:X31::, 2001:db8:K:4:X52::>. The SID behavior used in the example is End.X, as described in [RFC8986], but the procedure is equally applicable to any other (transit) SID type.

Figure 2 contains sample output for a ping request initiated at node N1 to a loopback address of node N5 via segment list <2001:db8:K:2:X31::, 2001:db8:K:4:X52::>.

```
> ping 2001:db8:L:5:: via segment list 2001:db8:K:2:X31::,  
    2001:db8:K:4:X52::
```

```
Sending 5, 100-byte ICMPv6 Echos to B5::, timeout is 2 seconds:  
!!!!  
Success rate is 100 percent (5/5), round-trip min/avg/max = 0.625  
/0.749/0.931 ms
```

Figure 2: Sample Ping Output at an SRv6-Capable Node

All transit nodes process the echo request message like any other data packet carrying an SR header and hence do not require any change. Similarly, the egress node does not require any change to process the ICMPv6 echo request. For example, in the example in Figure 2:

- \* Node N1 initiates an ICMPv6 ping packet with the SRH as follows:  
(2001:db8:L:1::, 2001:db8:K:2:X31::) (2001:db8:L:5::,  
2001:db8:K:4:X52::, 2001:db8:K:2:X31::, SL=2, NH = ICMPv6)(ICMPv6  
Echo Request).
- \* Node N2, which is an SRv6-capable node, performs the standard SRH processing. Specifically, it executes the End.X behavior indicated by the 2001:db8:K:2:X31:: SID and forwards the packet on link3 to node N3.
- \* Node N3, which is a non-SRv6-capable node, performs the standard IPv6 processing. Specifically, it forwards the echo request based on DA 2001:db8:K:4:X52:: in the IPv6 header.
- \* Node N4, which is an SRv6-capable node, performs the standard SRH processing. Specifically, it observes the End.X behavior (2001:db8:K:4:X52::) and forwards the packet on link10 towards node N5. If 2001:db8:K:4:X52:: is a PSP SID, the penultimate node (node N4) does not, should not, and cannot differentiate between the data packets and OAM probes. Specifically, if 2001:db8:K:4:X52:: is a PSP SID, node N4 executes the SID like any other data packet with DA = 2001:db8:K:4:X52:: and removes the SRH.
- \* The echo request packet at node N5 arrives as an IPv6 packet with or without an SRH. If node N5 receives the packet with an SRH, it skips SRH processing (SL=0). In either case, node N5 performs the standard ICMPv6 processing on the echo request and responds with the echo reply message to node N1. The echo reply message is IP routed.

#### A.1.2. Pinging a SID

The ping mechanism described above can also be used to perform SID reachability checks and to validate that the SID is locally programmed at the target node. This is explained in the following

example. The example uses ping to an End SID, as described in [RFC8986], but the procedure is equally applicable to ping any other SID behaviors.

Consider the example where the user wants to ping a remote SID 2001:db8:K:4::, via 2001:db8:K:2:X31::, from node N1. The ICMPv6 echo request is processed at the individual nodes along the path as follows:

- \* Node N1 initiates an ICMPv6 ping packet with the SRH as follows: (2001:db8:L:1::, 2001:db8:K:2:X31::) (2001:db8:K:4::, 2001:db8:K:2:X31::; SL=1; NH=ICMPv6)(ICMPv6 Echo Request).
- \* Node N2, which is an SRv6-capable node, performs the standard SRH processing. Specifically, it executes the End.X behavior indicated by the 2001:db8:K:2:X31:: SID on the echo request packet. If 2001:db8:K:2:X31:: is a PSP SID, node N4 executes the SID like any other data packet with DA = 2001:db8:K:2:X31:: and removes the SRH.
- \* Node N3, which is a non-SRv6-capable node, performs the standard IPv6 processing. Specifically, it forwards the echo request based on DA = 2001:db8:K:4:: in the IPv6 header.
- \* When node N4 receives the packet, it processes the target SID (2001:db8:K:4::).
- \* If the target SID (2001:db8:K:4::) is not locally instantiated and does not represent a local interface, the packet is discarded
- \* If the target SID (2001:db8:K:4::) is locally instantiated or represents a local interface, the node processes the Upper-Layer header. As part of the Upper-Layer header processing, node N4 responds to the ICMPv6 echo request message with an echo reply message. The echo reply message is IP routed.

## A.2. Traceroute in SRv6 Networks

The existing traceroute mechanisms, along the shortest path, continue to work without any modification. Any IPv6 node (SRv6-capable or a non-SRv6-capable) can initiate, transit, and egress a traceroute probe.

The following subsections outline some additional use cases of traceroute in SRv6 networks.

### A.2.1. Traceroute to an IPv6 Address via a Segment List

If an SRv6-capable ingress node wants to traceroute to an IPv6 address via an arbitrary segment list <S1, S2, S3>, it needs to initiate a traceroute probe with an SR header containing the SID list <S1, S2, S3>. The user issues a traceroute from node N1 to a loopback of node N5 via segment list <2001:db8:K:2:X31::, 2001:db8:K:4:X52::>. The SID behavior used in the example is End.X, as described in [RFC8986], but the procedure is equally applicable to any other (transit) SID type. Figure 3 contains sample output for

the traceroute request.

```
> traceroute 2001:db8:L:5:: via segment list 2001:db8:K:2:X31::,  
2001:db8:K:4:X52::
```

Tracing the route to 2001:db8:L:5::

```
1 2001:db8:2:1:21:: 0.512 msec 0.425 msec 0.374 msec  
   DA: 2001:db8:K:2:X31::,  
   SRH:(2001:db8:L:5::, 2001:db8:K:4:X52::, 2001:db8:K:2:X31::, SL=2)  
2 2001:db8:3:2:31:: 0.721 msec 0.810 msec 0.795 msec  
   DA: 2001:db8:K:4:X52::,  
   SRH:(2001:db8:L:5::, 2001:db8:K:4:X52::, 2001:db8:K:2:X31::, SL=1)  
3 2001:db8:4:3::41:: 0.921 msec 0.816 msec 0.759 msec  
   DA: 2001:db8:K:4:X52::,  
   SRH:(2001:db8:L:5::, 2001:db8:K:4:X52::, 2001:db8:K:2:X31::, SL=1)  
4 2001:db8:5:4::52:: 0.879 msec 0.916 msec 1.024 msec  
   DA: 2001:db8:L:5::
```

Figure 3: Sample Traceroute Output at an SRv6-Capable Node

In the sample traceroute output, the information displayed at each hop is obtained using the contents of the "Time Exceeded" or "Destination Unreachable" ICMPv6 responses. These ICMPv6 responses are IP routed.

In the sample traceroute output, the information for link3 is returned by node N3, which is a non-SRv6-capable node. Nonetheless, the ingress node is able to display SR header contents as the packet travels through the non-SRv6-capable node. This is because the "Time Exceeded" ICMPv6 message can contain as much of the invoking packet as possible without the ICMPv6 packet exceeding the minimum IPv6 MTU [RFC4443]. The SR header is included in these ICMPv6 messages initiated by the non-SRv6-capable transit nodes that are not running SRv6 software. Specifically, a node generating an ICMPv6 message containing a copy of the invoking packet does not need to understand the extension header(s) in the invoking packet.

The segment list information returned for the first hop is returned by node N2, which is an SRv6-capable node. Just like for the second hop, the ingress node is able to display SR header contents for the first hop.

There is no difference in processing of the traceroute probe at an SRv6-capable and a non-SRv6-capable node. Similarly, both SRv6-capable and non-SRv6-capable nodes may use the address of the interface on which probe was received as the source address in the ICMPv6 response. ICMPv6 extensions defined in [RFC5837] can be used to display information about the IP interface through which the datagram would have been forwarded had it been forwardable, the IP next hop to which the datagram would have been forwarded, the IP interface upon which the datagram arrived, and the sub-IP component of an IP interface upon which the datagram arrived.

The IP address of the interface on which the traceroute probe was received is useful. This information can also be used to verify if SIDs 2001:db8:K:2:X31:: and 2001:db8:K:4:X52:: are executed correctly

by nodes N2 and N4, respectively. Specifically, the information displayed for the second hop contains the incoming interface address 2001:db8:2:3:31:: at node N3. This matches the expected interface bound to End.X behavior 2001:db8:K:2:X31:: (link3). Similarly, the information displayed for the fourth hop contains the incoming interface address 2001:db8:4:5::52:: at node N5. This matches the expected interface bound to the End.X behavior 2001:db8:K:4:X52:: (link10).

### A.2.2. Traceroute to a SID

The mechanism to traceroute an IPv6 address via a segment list described in the previous section can also be used to traceroute a remote SID behavior, as explained in the following example. The example uses traceroute to an End SID, as described in [RFC8986], but the procedure is equally applicable to tracerouting any other SID behaviors.

Please note that traceroute to a SID is exemplified using UDP probes. However, the procedure is equally applicable to other implementations of traceroute mechanism. The UDP encoded message to traceroute a SID would use the UDP ports assigned by IANA for "traceroute use".

Consider the example where the user wants to traceroute a remote SID 2001:db8:K:4::, via 2001:db8:K:2:X31::, from node N1. The traceroute probe is processed at the individual nodes along the path as follows:

- \* Node N1 initiates a traceroute probe packet as follows (2001:db8:L:1::, 2001:db8:K:2:X31::) (2001:db8:K:4::, 2001:db8:K:2:X31::; SL=1; NH=UDP)(Traceroute probe). The first traceroute probe is sent with the hop-count value set to 1. The hop-count value is incremented by 1 for each subsequent traceroute probe.
- \* When node N2 receives the packet with hop-count = 1, it processes the hop-count expiry. Specifically, node N2 responds with the ICMPv6 message with type "Time Exceeded" and code "hop limit exceeded in transit". The ICMPv6 response is IP routed.
- \* When node N2 receives the packet with hop-count > 1, it performs the standard SRH processing. Specifically, it executes the End.X behavior indicated by the 2001:db8:K:2:X31:: SID on the traceroute probe. If 2001:db8:K:2:X31:: is a PSP SID, node N2 executes the SID like any other data packet with DA = 2001:db8:K:2:X31:: and removes the SRH.
- \* When node N3, which is a non-SRv6-capable node, receives the packet with hop-count = 1, it processes the hop-count expiry. Specifically, node N3 responds with the ICMPv6 message with type "Time Exceeded" and code "Hop limit exceeded in transit". The ICMPv6 response is IP routed.
- \* When node N3, which is a non-SRv6-capable node, receives the packet with hop-count > 1, it performs the standard IPv6 processing. Specifically, it forwards the traceroute probe based on DA 2001:db8:K:4:: in the IPv6 header.

- \* When node N4 receives the packet with DA set to the local SID 2001:db8:K:4::, it processes the End SID.
- \* If the target SID (2001:db8:K:4::) is not locally instantiated and does not represent a local interface, the packet is discarded.
- \* If the target SID (2001:db8:K:4::) is locally instantiated or represents a local interface, the node processes the Upper-Layer header. As part of the Upper-Layer header processing, node N4 responds with the ICMPv6 message with type "Destination Unreachable" and code "Port Unreachable". The ICMPv6 response is IP routed.

Figure 4 displays a sample traceroute output for this example.

```
> traceroute 2001:db8:K:4:X52:: via segment list 2001:db8:K:2:X31::

Tracing the route to SID 2001:db8:K:4:X52::
 1  2001:db8:2:1:21:: 0.512 msec 0.425 msec 0.374 msec
    DA: 2001:db8:K:2:X31::,
    SRH:(2001:db8:K:4:X52::, 2001:db8:K:2:X31::; SL=1)
 2  2001:db8:3:2:21:: 0.721 msec 0.810 msec 0.795 msec
    DA: 2001:db8:K:4:X52::,
    SRH:(2001:db8:K:4:X52::, 2001:db8:K:2:X31::; SL=0)
 3  2001:db8:4:3:41:: 0.921 msec 0.816 msec 0.759 msec
    DA: 2001:db8:K:4:X52::,
    SRH:(2001:db8:K:4:X52::, 2001:db8:K:2:X31::; SL=0)
```

Figure 4: Sample Output for Hop-by-Hop Traceroute to a SID

### A.3. Hybrid OAM Using the OAM Flag

This section illustrates a hybrid OAM mechanism using the 0-flag. Without loss of the generality, the illustration assumes node N100 is a centralized controller.

This illustration is different from the "in situ OAM" defined in [RFC9197]. This is because in situ OAM records operational and telemetry information in the packet as the packet traverses a path between two points in the network [RFC9197]. The illustration in this subsection does not require the recording of OAM data in the packet.

The illustration does not assume any formats for exporting the data elements or the data elements that need to be exported. The illustration assumes system clocks among all nodes in the SR domain are synchronized.

Consider the example where the user wants to monitor sampled IPv4 VPN 999 traffic going from CE1 to CE2 via a low-latency SR Policy P installed at node N1. To exercise a low-latency path, the SR Policy P forces the packet via segments 2001:db8:K:2:X31:: and 2001:db8:K:4:X52::. The VPN SID at node N7 associated with VPN 999 is 2001:db8:K:7:DT999::. 2001:db8:K:7:DT999:: is a USP SID. Nodes N1, N4, and N7 are capable of processing the 0-flag, but node N2 is

not capable of processing the 0-flag. Node N100 is the centralized controller capable of processing and correlating the copy of the packets sent from nodes N1, N4, and N7. Node N100 is aware of 0-flag processing capabilities. Node N100, with help from nodes N1, N4, and N7, implements a hybrid OAM mechanism using the 0-flag as follows:

- \* A packet P1 is sent from CE1 to node N1. The packet is:

P1: (IPv4 header)(payload)

- \* Node N1 steers packet P1 through the SR Policy P. Based on local configuration, node N1 also implements logic to sample traffic steered through SR Policy P for hybrid OAM purposes. Specification for the sampling logic is beyond the scope of this document. Consider the case where packet P1 is classified as a packet to be monitored via the hybrid OAM. Node N1 sets the 0-flag during the encapsulation required by SR Policy P. As part of setting the 0-flag, node N1 also sends a timestamped copy of packet P1 to a local OAM process. The packet is:

P1: (2001:db8:L:1::, 2001:db8:K:2:X31::) (2001:db8:K:7:DT999::, 2001:db8:K:4:X52::, 2001:db8:K:2:X31::; SL=2; 0-flag=1; NH=IPv4)(IPv4 header)(payload)

The local OAM process sends a full or partial copy of packet P1 to node N100. The OAM process includes the recorded timestamp, additional OAM information (like incoming and outgoing interface), and any applicable metadata. Node N1 forwards the original packet towards the next segment 2001:db8:K:2:X31::.

- \* When node N2 receives the packet with the 0-flag set, it ignores the 0-flag. This is because node N2 is not capable of processing the 0-flag. Node N2 performs the standard SRv6 SID and SRH processing. Specifically, it executes the End.X behavior [RFC8986] indicated by the 2001:db8:K:2:X31:: SID and forwards packet P1 over link3 towards node N3. The packet is:

P1: (2001:db8:L:1::, 2001:db8:K:4:X52::) (2001:db8:K:7:DT999::, 2001:db8:K:4:X52::, 2001:db8:K:2:X31::; SL=1; 0-flag=1; NH=IPv4)(IPv4 header)(payload)

- \* When node N3, which is a non-SRv6-capable node, receives packet P1, it performs the standard IPv6 processing. Specifically, it forwards packet P1 based on DA 2001:db8:K:4:X52:: in the IPv6 header.

- \* When node N4 receives packet P1, it processes the 0-flag. The packet is:

P1: (2001:db8:L:1::, 2001:db8:K:4:X52::) (2001:db8:K:7:DT999::, 2001:db8:K:4:X52::, 2001:db8:K:2:X31::; SL=1; 0-flag=1; NH=IPv4)(IPv4 header)(payload)

As part of processing the 0-flag, it sends a timestamped copy of the packet to a local OAM process. Based on local configuration, the local OAM process sends a full or partial copy of packet P1 to



node N100. The OAM process includes the recorded timestamp, additional OAM information (like incoming and outgoing interface, etc.), and any applicable metadata. Node N4 performs the standard SRv6 SID and SRH processing on the original packet P1. Specifically, it executes the End.X behavior indicated by the 2001:db8:K:4:X52:: SID and forwards packet P1 over link10 towards node N5. The packet is:

P1: (2001:db8:L:1::, 2001:db8:K:7:DT999::) (2001:db8:K:7:DT999::, 2001:db8:K:4:X52::, 2001:db8:K:2:X31::; SL=0; 0-flag=1; NH=IPv4)(IPv4 header)(payload)

- \* When node N5, which is a non-SRv6-capable node, receives packet P1, it performs the standard IPv6 processing. Specifically, it forwards the packet based on DA 2001:db8:K:7:DT999:: in the IPv6 header.
- \* When node N7 receives packet P1, it processes the 0-flag. The packet is:

P1: (2001:db8:L:1::, 2001:db8:K:7:DT999::) (2001:db8:K:7:DT999::, 2001:db8:K:4:X52::, 2001:db8:K:2:X31::; SL=0; 0-flag=1; NH=IPv4)(IPv4 header)(payload)

As part of processing the 0-flag, it sends a timestamped copy of the packet to a local OAM process. The local OAM process sends a full or partial copy of packet P1 to node N100. The OAM process includes the recorded timestamp, additional OAM information (like incoming and outgoing interface, etc.), and any applicable metadata. Node N7 performs the standard SRv6 SID and SRH processing on the original packet P1. Specifically, it executes the VPN SID indicated by the 2001:db8:K:7:DT999:: SID and, based on lookup in table 100, forwards packet P1 towards CE2. The packet is:

P1: (IPv4 header)(payload)

- \* Node N100 processes and correlates the copy of the packets sent from nodes N1, N4, and N7 to find segment-by-segment delays and provide other hybrid OAM information related to packet P1. For segment-by-segment delay computation, it is assumed that clocks are synchronized across the SR domain.
- \* The process continues for any other sampled packets.

#### A.4. Monitoring of SRv6 Paths

In the recent past, network operators demonstrated interest in performing network OAM functions in a centralized manner. [RFC8403] describes such a centralized OAM mechanism. Specifically, [RFC8403] describes a procedure that can be used to perform path continuity checks between any nodes within an SR domain from a centralized monitoring system. However, while [RFC8403] focuses on SR networks with MPLS data plane, this document describes how the concept can be used to perform path monitoring in an SRv6 network from a centralized controller.

In the reference topology in Figure 1, node N100 uses an IGP protocol like OSPF or IS-IS to get a view of the topology within the IGP domain. Node N100 can also use BGP-LS to get the complete view of an inter-domain topology. The controller leverages the visibility of the topology to monitor the paths between the various endpoints.

Node N100 advertises an End SID [RFC8986] 2001:db8:K:100:1::. To monitor any arbitrary SRv6 paths, the controller can create a loopback probe that originates and terminates on node N100. To distinguish between a failure in the monitored path and loss of connectivity between the controller and the network, node N100 runs a suitable mechanism to monitor its connectivity to the monitored network.

The following example illustrates loopback probes in which node N100 needs to verify a segment list <2001:db8:K:2:X31::, 2001:db8:K:4:X52::>:

- \* Node N100 generates an OAM packet (2001:db8:L:100::, 2001:db8:K:2:X31::)(2001:db8:K:100:1::, 2001:db8:K:4:X52::, 2001:db8:K:2:X31::, SL=2)(OAM Payload). The controller routes the probe packet towards the first segment, which is 2001:db8:K:2:X31::.
- \* Node N2 executes the End.X behavior indicated by the 2001:db8:K:2:X31:: SID and forwards the packet (2001:db8:L:100::, 2001:db8:K:4:X52::)(2001:db8:K:100:1::, 2001:db8:K:4:X52::, 2001:db8:K:2:X31::, SL=1)(OAM Payload) on link3 to node N3.
- \* Node N3, which is a non-SRv6-capable node, performs the standard IPv6 processing. Specifically, it forwards the packet based on DA 2001:db8:K:4:X52:: in the IPv6 header.
- \* Node N4 executes the End.X behavior indicated by the 2001:db8:K:4:X52:: SID and forwards the packet (2001:db8:L:100::, 2001:db8:K:100:1::)(2001:db8:K:100:1::, 2001:db8:K:4:X52::, 2001:db8:K:2:X31::, SL=0)(OAM Payload) on link10 to node N5.
- \* Node N5, which is a non-SRv6-capable node, performs the standard IPv6 processing. Specifically, it forwards the packet based on DA 2001:db8:K:100:1:: in the IPv6 header.
- \* Node N100 executes the standard SRv6 END behavior. It decapsulates the header and consumes the probe for OAM processing. The information in the OAM payload is used to detect missing probes, round-trip delay, etc.

The OAM payload type or the information carried in the OAM probe is a local implementation decision at the controller and is outside the scope of this document.

## Acknowledgements

The authors would like to thank Joel M. Halpern, Greg Mirsky, Bob Hinden, Loa Andersson, Gaurav Naik, Ketan Talaulikar, and Haoyu Song

for their review comments.

## Contributors

The following people contributed to this document:

Robert Raszuk  
Bloomberg LP  
Email: robert@raszuk.net

John Leddy  
Individual  
Email: john@leddy.net

Gaurav Dawra  
LinkedIn  
Email: gdawra.ietf@gmail.com

Bart Peirens  
Proximus  
Email: bart.peirens@proximus.com

Nagendra Kumar  
Cisco Systems, Inc.  
Email: naikumar@cisco.com

Carlos Pignataro  
Cisco Systems, Inc.  
Email: cpignata@cisco.com

Rakesh Gandhi  
Cisco Systems, Inc.  
Email: rgandhi@cisco.com

Frank Brockners  
Cisco Systems, Inc.  
Email: fbrockne@cisco.com

Darren Dukes  
Cisco Systems, Inc.  
Email: ddukes@cisco.com

Cheng Li  
Huawei  
Email: chengli13@huawei.com

Faisal Iqbal  
Individual  
Email: [faisal.ietf@gmail.com](mailto:faisal.ietf@gmail.com)

#### Authors' Addresses

Zafar Ali  
Cisco Systems  
Email: [zali@cisco.com](mailto:zali@cisco.com)

Clarence Filsfils  
Cisco Systems  
Email: [cfilsfil@cisco.com](mailto:cfilsfil@cisco.com)

Satoru Matsushima  
Softbank  
Email: [satoru.matsushima@g.softbank.co.jp](mailto:satoru.matsushima@g.softbank.co.jp)

Daniel Voyer  
Bell Canada  
Email: [daniel.voyer@bell.ca](mailto:daniel.voyer@bell.ca)

Mach(Guoyi) Chen  
Huawei  
Email: [mach.chen@huawei.com](mailto:mach.chen@huawei.com)