

SPRING
Internet-Draft
Intended status: Standards Track
Expires: 5 September 2022

C. Filsfils, Ed.
A. Abdelsalam, Ed.
P. Camarillo, Ed.
Cisco Systems, Inc.
M. Yufit
Broadcom
T. Graf
Swisscom
Y. Su
Alibaba, Inc
S. Matsushima
SoftBank
4 March 2022

Path Tracing in SRv6 networks
draft-filsfils-spring-path-tracing-00

Abstract

Path Tracing provides a record of the packet path as a sequence of interface ids. In addition, it provides a record of end-to-end delay, per-hop delay, and load on each egress interface along the packet delivery path.

Path Tracing allows to trace 14 hops with only a 40-bytes IPv6 Hop-by-Hop extension header.

Path Tracing supports fine grained timestamp. It has been designed for linerate hardware implementation in the base pipeline.

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 <https://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 5 September 2022.

Internet-Draft

Path Tracing

March 2022

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	2
2.	Terminology	3
2.1.	Requirements Language	4
3.	Midpoint Compressed Data	4
4.	PT Probing Instance	5
5.	PT Source Node Dataplane Behavior	6
6.	PT Midpoint Node Dataplane Behavior	7
7.	PT Sink Node Dataplane Behavior	7
8.	PT Headers	8
8.1.	IPv6 Hop-by-Hop Path Tracing Option	8
8.2.	SRH Path Tracing TLV	9
9.	Benefits	10
10.	Implementation Status	10
11.	Security Considerations	11
12.	IANA Considerations	12
12.1.	Destination Options and Hop-by-Hop Options	12
12.2.	Segment Routing Header TLV	12
13.	Acknowledgements	13
14.	Contributors	13
15.	References	13
15.1.	Normative References	13
15.2.	Informative References	14
	Authors' Addresses	15

[1.](#) Introduction

Path Tracing provides a record of the packet path as a sequence of

interface ids. In addition, it provides a record of end-to-end delay, per-hop delay, and load on each egress interface along the packet delivery path.

Path Tracing allows to trace 14 hops with only a 40 bytes IPv6 Hop-by-Hop header. The overhead is lower than [\[INT\]](#), [\[I-D.ietf-ippm-ioam-data\]](#), [\[I-D.song-opsawg-ifit-framework\]](#), and [\[I-D.kumar-ippm-ifa\]](#).

Path Tracing supports fine-grained timestamps. It has been designed for linerate hardware implementation in the base pipeline.

Path Tracing is applicable to both SR-MPLS [\[RFC8660\]](#), as well as SRv6 [\[RFC8986\]](#). This document defines the Path Tracing specification for the SRv6 dataplane. The SR-MPLS dataplane will be detailed in a separate document.

The specification proposed in this document has been demonstrated successfully in different interoperable hardware platforms at linerate ([Section 10](#)).

2. Terminology

The following terms used within this document are defined in [\[RFC8402\]](#), [\[RFC8754\]](#) and [\[RFC8986\]](#): Segment Routing (SR), SR Domain, Segment ID (SID), SRv6, SRv6 SID, SR Policy, Segment Routing Header (SRH), SR source node, transit node, SR Endpoint, SA, DA.

The following terms are used in this document as defined below:

PT: Path Tracing

MCD: Midpoint Compressed Data (MCD). Information that every transit router adds to the packet for PT purposes. Defined in [Section 3](#) of this document.

HbH-PT: IPv6 Hop-by-Hop [\[RFC8200\]](#) Path Tracing Option used for PT. It contains a stack of MCDs. It is defined in [Section 8.1](#) of this document

SRH PT-TLV: SRH TLV defined in [Section 8.2](#) of this document.

PT Source: A Source node that starts a PT Probing Instance (defined in [Section 4](#)) and generates PT probes.

PT Midpoint: A transit node that performs plain IPv6 forwarding (or SR Endpoint processing) and in addition records PT information in the HbH-PT.

PT Sink: A node that receives PT probes sent from the SRC containing the information recorded by every PT Midpoint along the path, and forwards them to a regional collector after recording its PT information.

RC: Regional collector that receives PT probes, parses, and stores them in TimeSeries Database. It uses the information in the HbH-PT and the SRH PT-TLV to construct the packet delivery path as well as the timestamp at each node.

[2.1.](#) 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.

[3.](#) Midpoint Compressed Data

Every PT Midpoint along the packet delivery path -from Source to Sink- records its PT information into the HbH-PT header. This information is known as Midpoint Compressed Data (MCD). It contains the following information:

- * MCD.OIF (Outgoing Interface ID): An 8-bit or 12-bit interface ID associated with the egress physical port of the router
 - The interface ID is assigned by an operator. The Interface IDs

are not globally unique across the entire network. Indeed the same Interface ID may be repeated multiple times in the network as long as the end-to-end path can be deterministically inferred based on the chain of Interface IDs.

- The programming of the Interface ID in the device may be done by CLI/NETCONF or any other means, and it is out of the scope of this document.
 - The usage of an 8-bit or 12-bit Interface ID is an operator choice, but the Interface ID size MUST be consistent across the entire network.
 - In case of Link Aggregation Groups (LAG/bundle) [[LAG](#)], each one of the members is configured with a different interface ID.
- * MCD.OIL (Outgoing Interface Load): A 4-bit representation of the egress interface load (i.e., current throughout relative to the port bandwidth).

- The load is represented using a 4-bit value in logarithmic scale. This allows more granular information as the load is higher.
- * MCD.TTS (Truncated Timestamp): An 8-bit timestamp encoding the time at which the packet egress the router.
- The 8-bit TTS has various possible significance depending on the link type. This is known as Time Template, and it is configured by the operator. For example, if the link is intercontinental, the 8-bit TTS encodes 7 bits of milliseconds and 1 bit of microseconds; whereas if the link is within an DC, the 8-bit TTS encodes 2 bits of milliseconds and 6 bits of microseconds.
 - Each egress port in the device is configured with one Time Template.
 - Note: all routers across the network MUST have time-synchronization. The mechanism used for time synchronization is out of the scope of this draft.

[4.](#) PT Probing Instance

The controller configures a PT Probing Instance at the source node. A PT Probing Instance is configured with the following parameters:

- * SA: the source address of the PT probe. Typically, it is the loopback address of the PT SRC.
- * Session ID: A 16-bit value.
- * Probe-rate: Number of probes per second to generate as part of this PT Probing Instance. The probe-rate is the aggregate of the probes generated across all the sweeping ranges.
- * SRv6 SID List: The SRv6 SID list associated with the packet. The last SID is the Sink node.
- * DSCP value
- * Hop-limit Value
- * IPv6 Flow-Label sweeping range:

- If set, different Flow-Label values must be used in the probe packets. It may be specified as a range of specific Flow-Label values to enumerate, or it may be specified as the number of different random Flow-Label values to use in a round-robin.
- * HbH-PT size
- * MTU sweeping range:
 - If set, payload must be included at the end of the packet to test different packet sizes.

[5.](#) PT Source Node Dataplane Behavior

For each configured PT Probing Instance, according to the probe-rate,

the PT SRC generates a PT probe packet as follows:

- S01. Generate a new IPv6 packet
- S02. Set the IPv6 SA as per PT Probing Instance configuration
- S03. Set the IPv6 DA to the first SID from the SRv6 SID List
- S04. Set the IPv6 Next Header field to 43 (SRH)
- S05. Set the DSCP and Flow Label values as per
PT Probing Instance configuration
- S06. Append an IPv6 Hop-by-Hop header with the Hop-by-Hop
Path Tracing option (HbH-PT)
- S07. Set all bits of the HbH-PT MCD Stack to zero
- S08. Append an SRH
- S09. Set the SRH Next Header field to 59 (IPv6 No Next Header)
- S10. Write the SID list in the SRH
- S11. Append the SRH PT-TLV
- S12. Add padding bytes after the SRH to reach the desired
packet size as per the MTU sweeping range configuration
- S13. Set the session ID field of the SRH PT-TLV as per
PT Probing Instance configuration
- S14. Set the Sequence Number field of SRH PT-TLV and
increase local counter
- S15. Perform an IPv6 FIB lookup to determine the Outgoing
Interface (IFACE-OUT) on which packet will be forwarded
- S16. Record Transmit 64-bit timestamp (SRC.T64) in the T64 field
of the SRH PT-TLV
- S17. Record IFACE-OUT ID (SRC.OIF) in the IF_ID field
of the SRH PT-TLV
- S18. Record IFACE-OUT Load (SRC.OIL) in the IF_LD field
of the SRH PT-TLV
- S19. Forward the packet via IFACE-OUT

Notes:

- * The pseudocode describes local processing at a node. An implementation of the pseudocode is compliant as long as the externally observable wire protocol is as described in the pseudocode.

[6.](#) PT Midpoint Node Dataplane Behavior

When a midpoint node receives an IPv6 packet that contains an IPv6

HbH-PT option, the node processes the HbH-PT as follows:

```
S01. When processing HbH-PT option {  
S02.   Compute the MCD information as per Section 3  
S03.   HbH-PT.MCD_Stack[MCD_Size:HbH-PT.OPT_Data_Len-1] =  
       HbH-PT.MCD_Stack[0:HbH-PT.OPT_Data_Len-(MCD_Size+1)]  
       //Shift HbH-PT MCD Stack to the right by MCD_Size bytes  
S04.   HbH-PT.MCD_Stack[0:MCD_Size-1] = MCD[0:MCD_Size-1]  
       //Push the MCD at the beginning of the Stack  
S05. }
```

Notes:

- * The PT Midpoint behavior MUST be implemented in the normal pipeline to experience the regular datapath (i.e., linerate). Offloading the processing of this option to either the slow-path or a co-processors is not acceptable and yields invalid results.

[7.](#) PT Sink Node Dataplane Behavior

We define a new SRv6 Endpoint Behavior called "Endpoint Behavior bound to an SRv6 Policy with Timestamp, Encapsulation and Forward" ("End.B6.TEF" for short).

It is a Binding SID instantiated, at Sink nodes, that encapsulates the packet with a new IPv6 header, an SRH that contains the SID list associated to End.B6.TEF SID and an SRH PT-TLV that is used to carry Path Tracing information of Sink node.

When N receives a packet whose IPv6 DA is S and S is a local End.B6.TEF SID, N does the following:

S01. Record Rx 64-bit timestamp (SNK.T64)

- o 00: Skip HbH for nodes that don't support the HbH-PT Option Type
 - o 1: update HbH-PT for nodes that support the HbH-PT Option Type
- Opt Data Len: the length of the MCD stack in bytes.

Note: The IPv6 Path Tracing Option has a variable length. It is RECOMMENDED that implementations support a 38-octet HbH-PT Option. The operator, upon configuring the Source node behavior, MUST select an option length that is supported by all the routers in the network.

8.2. SRH Path Tracing TLV

We define a new SRH TLV, called "Path Tracing TLV" ("SRH PT-TLV" for short). It has the following format:

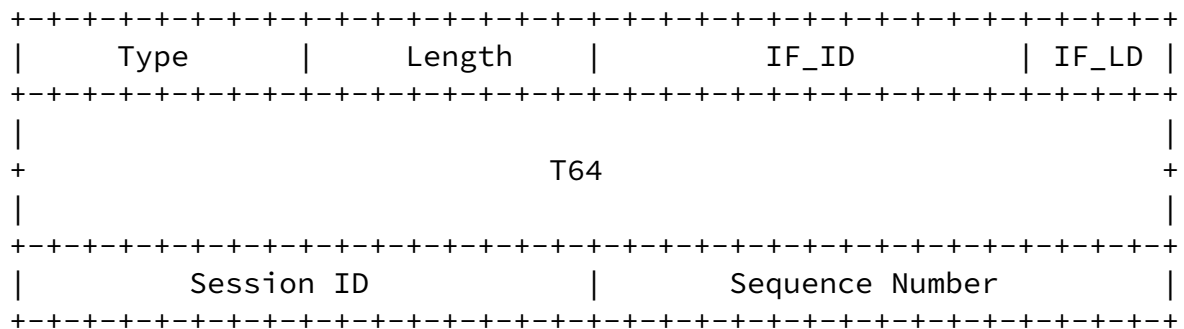


Figure 2: SRH Path Tracing TLV Format

Where:

- * Type: TBA2-1
- * Length: 14
- * IF_ID: 12-bit Interface ID
- * IF_LD: 4-bit Interface Load
- * T64: 64-bit PTP Timestamp
- * Session ID: Session identifier set by SRC node generating the probes. Used to co-relate probes of the same session. Value of zero means unset.

- * Sequence Number: the sequence number of the probe set by SRC node generating the probes. Value of zero means unset.

Note: The SRH PT-TLV is generated by both the PT SRC and the PT SNK. When used at the PT SNK node, the Session ID, and Sequence Number fields MUST be set to zero.

[9.](#) Benefits

- * Low overhead:
 - A 40Byte Hop-By-Hop header allows for 14 hops path measurements: 1 at the PT SRC, 12 at PT Midpoint routers and 1 at the PT SNK
 - PT has the lowest MTU overhead compared to alternative solutions such as [\[INT\]](#), [\[I-D.ietf-ippm-ioam-data\]](#), [\[I-D.song-opsawg-ifit-framework\]](#), and [\[I-D.kumar-ippm-ifa\]](#).
- * Linerate and HW friendliness:
 - Implemented at linerate in current hardware, using the regular forwarding pipeline. No offloading to co-processors or slow-path whose databases might defer from forwarding pipeline.
 - Leverages mature hardware capabilities (basic shift operation); no packet resizing at every node along the path
 - High number of diverse linerate interoperable hardware Implementations (see [Section 10](#))
- * Scalable Fine-grained Timestamp:
 - 64bit at PT SRC and PT SNK
 - 8bit at PT Midpoint leveraging flexible per-outgoing-link template allowing diverse link types in the same measurement (e.g., DC, metro, WAN)
- * Scalable Load measurement

[10.](#) Implementation Status

Editorial note: Please remove this section prior publication.

The following routing platforms have participated in an interop testing:

- * Cisco 8802 (based Cisco Silicon One Q200)
- * Cisco ASR9904 with Lightspeed linecard

Filsfils, et al.

Expires 5 September 2022

[Page 10]

Internet-Draft

Path Tracing

March 2022

- * Cisco NCS5508 (based on Broadcom Jericho2 platform)
- * Cisco Nexus N3K-C3464C (based on Barefoot Tofino)
- * Marvel Prestera Falcon

The following open-source software networking stacks have also participated in the interop:

- * FD.io VPP
- * Linux Kernel

The following opensource applications also have extensions to support Path Tracing:

- * Wireshark
- * Tcpdump
- * P4 implementation for software switch

11. Security Considerations

The security considerations for Segment Routing are discussed in [\[RFC8402\]](#). [Section 5 of \[RFC8754\]](#) describes the SR Deployment Model and the requirements for securing the SR Domain. The security considerations of [\[RFC8754\]](#) also cover topics such as attack vectors and their mitigation mechanisms that also apply to the behaviors introduced in this document. Together, they describe the required security mechanisms that allow establishment of an SR domain of trust. Having such a well-defined trust boundary is necessary in

order to operate SRv6-based services for internal traffic while preventing any external traffic from accessing or exploiting the SRv6-based services.

This document defines the Path Tracing architecture, which is deployed on a secured SRv6-domain. As such, all the security considerations defined in [[RFC8754](#)], [[RFC8402](#)], and [[RFC8986](#)] are applicable.

In addition, any border router in an SR Domain network where Path Tracing is enabled, MUST support the configuration of the following ACLs:

- * If there is a packet coming from an external interface destined towards an internal interface that contains an IPv6 Hop-by-Hop header with a Path Tracing option, then such packet is silently dropped.
- * If there is a packet coming from an internal interface destined towards an external interface that contains an IPv6 Hop-by-Hop header with a Path Tracing option, then such packet is silently dropped.

These ACLs SHOULD be enabled by default. An operator MAY disable them individually based on local configuration.

The processing of IPv6 Hop-by-Hop headers could sometimes be used as an attack vector to overload the CPU of the router. As defined in [Section 6](#) of this document, the HBH-PT option MUST be processed at line rate. Therefore there is no impact on the router's CPU.

[12.](#) IANA Considerations

This document has two actions for IANA:

[12.1.](#) Destination Options and Hop-by-Hop Options

This I-D requests IANA to allocate a new entry in the "Destination

Options and Hop-by-Hop Options" sub-registry under the top-level registry "Internet Protocol Version 6 (IPv6) Parameters":

Value	Description	Reference

TBA1-1	Path Tracing	[This.ID]

Note: The 3 high-order bits must be 001.

[12.2.](#) Segment Routing Header TLV

This I-D requests IANA to allocate a new entry in the "Segment Routing Header TLVs" sub-registry under the top-level registry "Internet Protocol Version 6 (IPv6) Parameters":

Value	Description	Reference

TBA2-1	Path Tracing TLV	[This.ID]

[13.](#) Acknowledgements

The authors of this document would like to thank the team that has collaborated on the design and implementation of the Path Tracing framework at Cisco, Broadcom, Marvel, Swisscom, Alibaba, Softbank, University of Rome "Tor Vergata", and ETH Zurich. In particular: Eyal Dagan, Guy Caspary, Elad Naor, Aviran Kadosh, Eli Stein, Oren Yabo, Aviad Behar, Anand Sridharan, Anju Dey, John Bettink, Kamran Raza, Asif Islam, Yue Gao, Jakub Horn, Sam Kheirallah, Shelly Cadora, Kris Michielsen, Francois Clad, Stefano Salsano, Andrea Mayer, Paolo Lungaroni, Giulio Sidoretti, Leonardo Rodoni, Marco Tollini.

[14.](#) Contributors

Jisu Bhattacharya, Cisco Systems; jisu@cisco.com

Rakesh Gandhi, Cisco Systems; rgandhi@cisco.com

Serguei Bezverkhi, Cisco Systems; sbezverk@cisco.com

Sonia Ben Ayed, Cisco Systems; sbenayed@cisco.com

Israel Meilik, Broadcom; israel.meilik@broadcom.com

Shay Zadok, Broadcom; shay.zadok@broadcom.com

Weiqiang Cheng, China Mobile; chengweiqiang@chinamobile.com

15. References

15.1. Normative References

- [RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, [RFC 8200](#), DOI 10.17487/RFC8200, July 2017, <<https://www.rfc-editor.org/info/rfc8200>>.
- [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>>.

Filsfils, et al.

Expires 5 September 2022

[Page 13]

Internet-Draft

Path Tracing

March 2022

- [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>>.

15.2. Informative References

- [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>>.
- [RFC8660] Bashandy, A., Ed., Filsfils, C., Ed., Previdi, S., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing with the MPLS Data Plane", [RFC 8660](#), DOI 10.17487/RFC8660, December 2019, <<https://www.rfc-editor.org/info/rfc8660>>.
- [I-D.ietf-ippm-ioam-data] Brockners, F., Bhandari, S., and T. Mizrahi, "Data Fields for In-situ OAM", Work in Progress, Internet-Draft, [draft-ietf-ippm-ioam-data-17](#), 13 December 2021, <<https://www.ietf.org/archive/id/draft-ietf-ippm-ioam-data-17.txt>>.
- [I-D.kumar-ippm-ifa] Kumar, J., Anubolu, S., Lemon, J., Manur, R., Holbrook, H., Ghanwani, A., Cai, D., Ou, H., Li, Y., and X. Wang, "Inband Flow Analyzer", Work in Progress, Internet-Draft, [draft-kumar-ippm-ifa-04](#), 20 January 2022, <<https://www.ietf.org/archive/id/draft-kumar-ippm-ifa-04.txt>>.
- [I-D.song-opsawg-ifit-framework] Song, H., Qin, F., Chen, H., Jin, J., and J. Shin, "A Framework for In-situ Flow Information Telemetry", Work in Progress, Internet-Draft, [draft-song-opsawg-ifit-framework-17](#), 22 February 2022, <<https://www.ietf.org/archive/id/draft-song-opsawg-ifit-framework-17.txt>>.
- [INT] "In-band Network Telemetry (INT) Dataplane Specification", 2020, <https://github.com/p4lang/p4-applications/blob/master/docs/INT_v2_1.pdf>.

- [LAG] "IEEE Standard for Local and metropolitan area networks - Link aggregation, IEEE std 802.1AX-2008", IEEE , 2008.

Clarence Filsfils (editor)
Cisco Systems, Inc.
Belgium
Email: cf@cisco.com

Ahmed Abdelsalam (editor)
Cisco Systems, Inc.
Italy
Email: ahabdels@cisco.com

Pablo Camarillo Garvia (editor)
Cisco Systems, Inc.
Spain
Email: pcamaril@cisco.com

Mark Yufit
Broadcom
Israel
Email: mark.yufit@broadcom.com

Thomas Graf
Swisscom
Switzerland
Email: thomas.graf@swisscom.com

Yuanchao Su
Alibaba, Inc
China
Email: yitai.syc@alibaba-inc.com

Satoru Matsushima
SoftBank
Japan
Email: satoru.matsushima@g.softbank.co.jp