Workgroup: SPRING
Internet-Draft:
draft-filsfils-spring-path-tracing-03
Published: 13 February 2023
Intended Status: Standards Track
Expires: 17 August 2023
Authors: C. Filsfils Cisco Systems, Inc.
A. Abdelsalam, Ed. Cisco Systems, Inc.
P. Camarillo, Ed. Cisco Systems, Inc.
M. Yufit Broadcom
T. Graf Swisscom
Y. Su Alibaba, Inc
S. Matsushima SoftBank
M. Valentine Goldman Sachs
A. Dhamija Rakuten

# Path Tracing in SRv6 networks

## 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

## Table of Contents



## 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], [RFC9197], [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 implemented successfully in different interoperable hardware platforms at linerate (Section 11).

## 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 9.1 of this document

SRH PT-TLV: SRH TLV defined in Section 9.2 of this document.

PT Source: A Source node that starts a PT Probing Instance (defined in Section 5) 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 interface 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 interface 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.

  - Each egress interface in the device is configured with a TTS template.

  - The TTS template defines the position of 8-bits to be selected from the egress timestamp. Section 4 of this document discusses the timestamp format used in path tracing.

  - A Path Tracing Midpoint implementation MAY support one or more TTS templates. Each TTS template provides a different time precision.

  - An operator configures an egress interface with a single TTS template. The choice of the TTS template for a given interface is based on the type of the link connected to that interface. For example, an interface connected to DC link will have a different TTS Template from an interface connected to intercontinental or WAN link, as they have different precision requirements.

## 4. Timestamp requirements

### 4.1. Timestamp format

Path Tracing uses a 64-bit timestamp format. [RFC8877] recommends two 64-bit timestamp formats: 64-bit Truncated PTP timestamp format and NTP 64-bit timestamp format. Path Tracing can work with both formats indifferently.

### 4.2. Time synchronization

All routers across the network MUST have time-synchronization. PTP [IEEE1588] and NTP [RFC5905] are example protocols that can be used for time-synchronization.

## 5. 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.

## 6. 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 zero (HbH)
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.

## 7. 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 with full PPS and full BW). Offloading the processing of this option to either the slow-path or a co-processors is not acceptable and yields invalid results.

## 8. 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)
S02. Record incoming interface ID (SNK.IIF)
S03. Record incoming interface Load (SNK.IIL)
S04. Push a new IPv6 header
S05. Set the IPv6 SA to the Sink node loopback
S06. Set the IPv6 DA to the first SID in the SRv6 SID List
S07. Set the IPv6 Next Header field to 43 (SRH)
S08. Append an SRH
S09. Set the SRH Next Header field to 41 (IPv6)
S10. Write the SID list in the SRH
S11. Append the SRH PT-TLV
S12. Set the session ID field of the SRH PT-TLV to zero
S13. Set the Sequence Number field of the SRH PT-TLV to zero
S14. Write SNK.T64 in the T64 field of the SRH PT-TLV
S15. Write SNK.IIF in the IF_ID field of the SRH PT-TLV
S16. Write SNK.IIL in the IF_LD field of the SRH PT-TLV
S17. Submit the packet to the egress IPv6 FIB lookup for
        transmission to the new destination
```

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.

## 9. PT Headers

### 9.1. IPv6 Hop-by-Hop Path Tracing Option

This document defines a new IPv6 Path Tracing option to be carried in the IPv6 Hop-by-Hop Header. The option has the following format:

```
                                +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                                |  Option Type  |  Opt Data Len |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                          MCD  Stack                           ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: IPv6 Hop-by-Hop Path Tracing Option Format

Where:

*Option Type: TBA1-1

-The 3 high-order bits of the option must be set to 001

o00: Skip HbH for nodes that don't support the HbH-PT Option Type

o1: 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.

### 9.2. SRH Path Tracing TLV

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

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |     Length    |          IF_ID        | IF_LD |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                             T64                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Session ID            |       Sequence Number         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

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 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.

## 10. 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], [RFC9197], [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 11)

*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

## 11. 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

*Cisco NCS5508 (based on Broadcom Jericho2 platform)

*Cisco Nexus N3K-C3464C (based on Barefoot Tofino)

*Marvell Prestera Falcon

*Keysight IxNetwork

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

## 12. 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 7 of this document, the HBH-PT option MUST be processed in the router's fast path. Therefore, there is no impact on the router's CPU.

## 13. IANA Considerations

This document has two actions for IANA:

### 13.1. 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.

### 13.2. 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] |

## 14. 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, Keysight, 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, Yuanwen Sun, Anirban Bhattacharya, Ajay Ramamurthy.

## 15. References

### 15.1. 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>.

**[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/rfc/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/rfc/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/rfc/rfc8986>.

### 15.2. Informative References

**[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-05, 12 August 2022, <https://datatracker.ietf.org/doc/html/draft-kumar-ippm-ifa-05>.

**[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-19, 24 October 2022, <https://datatracker.ietf.org/doc/html/draft-song-opsawg-ifit-framework-19>.

**[IEEE1588]** "IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems", IEEE , 2008, <https://doi.org/10.1109/IEEESTD.2008.4579760>.

**[INT]** "In-band Network Telemetry (INT) Dataplane Specification", 2020, <https://github.com/p4lang/p4-applications/blob/master/docs/INT_v2_1.pdf>.

**[LAG]** "802.1AX-2014 - IEEE Standard for Local and metropolitan area networks -- Link Aggregation", IEEE , 2014, <https://doi.org/10.1109/IEEESTD.2014.7055197>.

**[RFC5905]** Mills, D., Martin, J., Ed., Burbank, J., and W. Kasch, "Network Time Protocol Version 4: Protocol and Algorithms Specification", RFC 5905, DOI 10.17487/RFC5905, June 2010, <https://www.rfc-editor.org/rfc/rfc5905>.

**[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/rfc/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/rfc/rfc8660>.

**[RFC8877]** Mizrahi, T., Fabini, J., and A. Morton, "Guidelines for Defining Packet Timestamps", RFC 8877, DOI 10.17487/RFC8877, September 2020, <https://www.rfc-editor.org/rfc/rfc8877>.

**[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/rfc/rfc9197>.

## Contributors

Jisu Bhattacharya
Cisco Systems, Inc.
United States of America

Email: jisu@cisco.com

Rakesh Gandhi
Cisco Systems, Inc.
Canada

Email: rgandhi@cisco.com

Serguei Bezverkhi
Cisco Systems, Inc.
Italy

Email: sbezverk@cisco.com

Sonia Ben Ayed
Cisco Systems, Inc.
France

Email: sbenayed@cisco.com

Israel Meilik
Broadcom
Israel

Email: israel.meilik@broadcom.com

Shay Zadok
Broadcom
Israel

Email: shay.zadok@broadcom.com

Weiqiang Cheng
China Mobile
China

Email: chengweiqiang@chinamobile.com

## Authors' Addresses

Clarence Filsfils
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

Mike Valentine
Goldman Sachs
United States of America

Email: michael.j.valentine@gs.com

Amit Dhamija
Rakuten
India

Email: amit.dhamija@rakuten.com