SPRING Working Group
Internet-Draft
Intended status: Informational
Expires: 4 October 2026

R. Gandhi, Ed.
C. Filsfils
Cisco Systems, Inc.
B. Janssens
Colt
M. Chen
Huawei
R. Foote
Nokia
2 April 2026

# Performance Measurement Using Simple Two-Way Active Measurement Protocol (STAMP) for Segment Routing over the IPv6 (SRv6) Data Plane
draft-ietf-spring-stamp-srpm-srv6-01

## Abstract

Segment Routing (SR) can be used to steer packets through a network employing source routing. SR can be applied to both MPLS (SR-MPLS) and IPv6 (SRv6) data planes. This document describes the procedures for Performance Measurement for SRv6 using the Simple Two-Way Active Measurement Protocol (STAMP), as defined in RFC 8762, along with its optional extensions defined in RFC 8972 and further augmented in RFC 9503. The described procedures are used for links and SRv6 paths (including Segment Lists of SRv6 Policies, SRv6 IGP best paths, and SRv6 IGP Flexible Algorithm paths), as well as Layer-3 and Layer-2 services over the SRv6 paths.

## Status of This Memo

## Copyright Notice



## Table of Contents

## 1. Introduction

Segment Routing (SR) [RFC8402] can be used to steer packets through a network employing source routing. SR can be applied to both MPLS (SR-MPLS) and IPv6 (SRv6) data planes. SR takes advantage of Equal-Cost Multipath (ECMP) between source and transit nodes, between transit nodes, and between transit and destination nodes. SR Policies, as defined in [RFC9256], are used to steer traffic through specific user-defined paths using a list of segments.

A comprehensive SR Performance Measurement toolset is an essential requirement for measuring network performance to provide Service Level Agreements (SLAs).

The Simple Two-Way Active Measurement Protocol (STAMP), as specified in [RFC8762], provides capabilities for measuring various performance metrics in IP networks without the use of a control channel to pre-signal session parameters. [RFC8972] defines optional extensions in the form of TLVs for STAMP. [RFC9503] further augments that framework to define STAMP extensions for SR networks.

This document describes the procedures for Performance Measurement in SRv6 networks, using STAMP as defined in [RFC8762], along with its optional extensions defined in [RFC8972] and augmented in [RFC9503]. The described procedures are used for links and SRv6 paths [RFC8402] (including Segment Lists of SRv6 Policies [RFC9256], SRv6 IGP best paths, and SRv6 Flexible Algorithm (Flex-Algo) paths [RFC9350]), as well as Layer-3 (L3) and Layer-2 (L2) services over the SRv6 paths.

STAMP requires protocol support on the Session-Reflector to process the received test packets. As a result, the received test packets need to be punted from the fast path in the data plane, and return test packets need to be generated. This limits the frequency of STAMP test packets and the ability to provide faster measurement intervals. This document adds new mechanisms to enhance the procedures for Performance Measurement using STAMP to improve the scalability of the number of STAMP sessions and the measurement interval for SRv6 paths by defining new measurement modes: one-way, loopback, and loopback with "timestamp and forward."

# 2. Conventions Used in This Document

## 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.

## 2.2. Abbreviations

ECMP: Equal Cost Multi-Path.

HMAC: Hashed Message Authentication Code.

L2: Layer-2.

L3: Layer-3.

MBZ: Must be Zero.

PSID: Path Segment Identifier.

SHA: Secure Hash Algorithm.

SID: Segment ID.

SR: Segment Routing.

SRH: Segment Routing Header.

SRv6: Segment Routing over the IPv6 data plane.

SSID: STAMP Session Identifier.

STAMP: Simple Two-Way Active Measurement Protocol.

TC: Traffic Class.

TSF: Timestamp and Forward.

VPN: Virtual Private Network.

## 3. Overview

For performance measurement in SRv6 networks, the STAMP Session-Sender and Session-Reflector use the STAMP test packets defined in [RFC8762], along with optional extensions defined in [RFC8972]. The STAMP test packets are encapsulated using an IP/UDP header, as specified in [RFC8762]. In this document, the STAMP test packets using the IP/UDP header are used for SRv6 networks, where the STAMP test packets are further encapsulated with an IPv6 Segment Routing Header (IPv6/SRH).

STAMP test packets are transmitted in performance measurement modes, including two-way, one-way, loopback, and loopback with "timestamp and forward" in SRv6 networks. Note that the two-way measurement mode is referenced in the STAMP process in [RFC8762] and is further described for SRv6 networks in this document. The other measurement modes, which are new and specifically described for SRv6 networks in this document, are not defined by the STAMP process in [RFC8762].

STAMP test packets are transmitted on the same path as the data traffic flow under measurement to measure the delay and packet loss experienced by the data traffic flow, using the same IPv6/SRH encapsulation as the data traffic flow. Similarly, STAMP test packets are transmitted on various transport data paths in the network to measure the delay and packet loss experienced by the traffic forwarded on those transport data paths. The STAMP test packets carry the same IPv6/SRH headers as the data packets transmitted on the SRv6 path and as the data packets forwarded over the L3 and L2 services over the SRv6 paths.

For encapsulating the STAMP test packets for the SRv6 data plane, two modes of encoding are defined in this document: Insert-Mode and Encaps-Mode.  In Insert-Mode, an SRH is inserted after the IPv6 header of the test packets.  In Encaps-Mode, the test packets with an IP header are further encapsulated with an outer IPv6/SRH.  The Session-Sender generates the STAMP test packets locally in either of the two encapsulation modes, based on local provisioning.

Typically, STAMP reply test packets are transmitted along an IP path between the Session-Reflector and Session-Sender.  Matching the forward direction path and return path for STAMP test packets, even for directly connected nodes, is not guaranteed.  In SRv6 networks, it may be desired that the same path (i.e., the same set of links and nodes) between the Session-Sender and Session-Reflector be used for the STAMP test packets in both directions, for example, in an ECMP environment.

In two-way measurement mode, this is achieved by using the optional STAMP extensions for SRv6 networks, as specified in [RFC9503].  The STAMP Session-Reflector uses the return path parameters for the reply test packet from the STAMP extensions in the received Session-Sender test packet, as described in [RFC9503].  In loopback measurement mode, this is achieved by adding both the forward direction path and the return path in the IPv6/SRH encapsulation of the Session-Sender test packets.

The performance measurement procedures defined in this document are used to measure both delay and packet loss in SRv6 networks based on the transmission and reception of STAMP test packets.  The optional STAMP extensions, as defined in [RFC8972], are used for direct measurement in SRv6 networks.

The compression of an SRv6 Segment List as specified in [RFC9800] is equally applicable to the performance measurement procedures defined in this document to take advantage of significantly reducing the size of the SRv6 encapsulation needed to transmit STAMP test packets over long Segment Lists.

### 3.1.  STAMP Reference Model

The STAMP Reference Model, along with some typical measurement parameters, as defined in [RFC8972] for a STAMP session, is shown in Figure 1.

```
                               +------------+
                               |    SDN     |
                               | Controller |
                               +------------+
                                    /  \
     Performance Measurement Mode  /    \         Stateful or Stateless
     Destination UDP Port         /      \        Destination UDP Port
     Authentication Mode         /        \       Authentication Mode
         Keychain               /          \          Keychain
     Timestamp Format          /            \     Timestamp Format
     Metric Type              /              \
     SSID                    /                \
                            v                  v
                        +-------+          +-------+
                        |       |  STAMP   |       |
                        |   S1  |==========|   R1  |
                        |       |  Session |       |
                        +-------+          +-------+

                  STAMP Session-Sender  STAMP Session-Reflector
```

Figure 1: STAMP Reference Model

The procedure, as defined in [RFC8972], uses the two-way measurement mode.

The destination UDP port number is selected for the STAMP function as described in [RFC8762]. By default, the reflector UDP port 862 is selected as the destination UDP port for STAMP sessions [RFC8762] for links, SRv6 paths, and L3 and L2 services over the SRv6 paths.

The source UDP port is selected by the Session-Sender. The same or different source UDP ports may be used for different STAMP sessions.

Session-Reflector mode can be either Stateful or Stateless, as described in Section 4 of [RFC8762]. Stateless Session-Reflector mode is applicable only in two-way measurement mode.

The SSID field in the STAMP test packets [RFC8972], along with the local configuration, is used to identify the STAMP sessions.

When authentication mode is enabled for STAMP sessions, the matching Authentication Type (e.g., HMAC-SHA-256) and Keychain must be configured on both the Session-Sender and Session-Reflector [RFC8762].

Examples of the Timestamp Format include 64-bit truncated Precision Time Protocol (PTPv2) [IEEE.1588] and 64-bit Network Time Protocol (NTPv4) [RFC5905]. By default, the Session-Reflector replies using the same timestamp format as received in the Session-Sender test packet, as indicated by the "Z" flag in the Error Estimate field, as described in [RFC8762]. This behaviour can be based on the Session-Reflector's capability.

Examples of Delay Metrics are one-way delay, round-trip delay, near-end delay (forward direction), and far-end delay (backward direction), as defined in [RFC8762].

Examples of Packet Loss Metric Type are round-trip packet loss, near-end packet loss (forward direction) and far-end packet loss (backward direction), as defined in [RFC8762].

A Software-Defined Networking (SDN) controller can be used for the configuration and management of STAMP sessions, as described in [RFC8762]. The controller can also receive streaming telemetry of operational data. The YANG data model for STAMP, defined in [I-D.ietf-ippm-stamp-yang], can be used to configure Session-Senders and Session-Reflectors and to stream telemetry of operational data.

## 4. Two-Way Measurement Mode

As shown in Figure 2, the reference topology for two-way measurement mode, the STAMP Session-Sender S1 initiates a STAMP Session-Sender test packet, and the STAMP Session-Reflector R1 generates and transmits a reply test packet. The reply test packets are transmitted to the STAMP Session-Sender S1 on the same path (i.e., the same set of links and nodes) or on a different path in the reverse direction from the path taken towards the Session-Reflector R1.

T1 is a transmit timestamp, and T4 is a receive timestamp added by node S1. T2 is a receive timestamp, and T3 is a transmit timestamp added by node R1. All four timestamps are used by the Session-Sender to measure the round-trip delay metric as ((T4 - T1) - (T3 - T2)). Timestamps T1 and T2 are used by the Session-Sender to measure the one-way delay metric as (T2 - T1), also referred to as the near-end (forward direction) delay metric. Note that the delay value (T4 - T3), measured by the Session-Sender, is referred to as the far-end (backward direction) one-way delay metric.

The computation of the one-way delay metric requires the clocks on the Session-Sender and Session-Reflector to be synchronized using either PTPv2 or NTPv4.

```
                          T1                T2
                         /                    \
                +-------+     Test Packet     +-------+
                |       | - - - - - - - - - ->|       |
                |   S1  |=====================|   R1  |
                |       |<- - - - - - - - - - |       |
                +-------+  Reply Test Packet  +-------+
                         \                    /
                          T4                T3

          STAMP Session-Sender          STAMP Session-Reflector
```

Figure 2: Reference Topology for Two-Way Measurement Mode

The nodes S1 and R1 may be connected via a link or an SRv6 path [RFC8402]. The link can be a physical interface, a virtual link, a Link Aggregation Group (LAG) [IEEE802.1AX], or a LAG member link. The SRv6 path may be a Segment List of an SRv6 Policy [RFC9256] on node S1 (referred to as the "head-end") with a destination to node R1 (referred to as the "endpoint"), an SRv6 IGP best path, or an SRv6 IGP Flex-Algo path [RFC9350]. Additionally, a Layer-3 (L3) or Layer-2 (L2) VPN service may be carried over the SRv6 path between nodes S1 and R1.

### 4.1. Session-Sender Test Packet

The content of a Session-Sender test packet is shown in Figure 3. The payload containing the Session-Sender test packet, as defined in Section 3 of [RFC8972], is transmitted with an IP and UDP header [RFC0768].

```
    +---------------------------------------------------------------+
    | IP Header                                                     |
    .  Source IP Address = Session-Sender IP Address                .
    .  Destination IP Address = Session-Reflector IP Address        .
    .  IPv4 Protocol or IPv6 Next-header = 17 (UDP)                 .
    .                                                               .
    +---------------------------------------------------------------+
    | UDP Header                                                    |
    .  Source Port = Chosen by Session-Sender                       .
    .  Destination Port = User-configured Destination Port | 862    .
    .                                                               .
    +---------------------------------------------------------------+
    | Payload = Test Packet as specified in Section 3 of RFC 8972   |
    .           in Figures 1 and 3                                  .
    .                                                               .
    +---------------------------------------------------------------+
```

Figure 3: Content of Session-Sender Test Packet

### 4.2. Session-Sender Test Packet for Links

The Session-Sender test packet, as shown in Figure 3, is transmitted over the link for delay measurement. The local and remote IP addresses of the link are used as the Source and Destination Addresses in the IP header of the Session-Sender test packet, respectively. For IPv6 links, the link-local address [RFC7404] may also be used in the IP header.

The Session-Sender uses a discovery protocol or other means to discover the peer IP and MAC addresses for the links. For example, the Session-Sender can use the Address Resolution Protocol (ARP) or the Neighbour Discovery Protocol (NDP) table to obtain the IP and MAC addresses for the links when transmitting STAMP packets.

Note that the Session-Sender test packet is further encapsulated with a Layer-2 header containing the Session-Reflector MAC address as the Destination MAC address and the Session-Sender MAC address as the Source MAC address for Ethernet links.

For delay measurement of LAG member links, a separate STAMP micro-session is created for each member of the LAG. The STAMP extension for the Micro-Session ID TLV, as defined in [RFC9534], is used to identify each member link of the LAG associated with the STAMP micro-session on the Session-Sender and Session-Reflector. The Session-Reflector replies on the same member of the LAG in the reverse direction, based on the received Session-Sender test packet and on either the local configuration or the received information from the data plane.

### 4.3. Session-Sender Test Packet for SRv6 Data Plane

The Session-Sender generates the STAMP test packets for the SRv6 data plane, which can be encoded in either Encaps-Mode or Insert-Mode.

When the Session-Sender test packets are encoded in Encaps-Mode, the test packets are generated with the IP header, and the outer IPv6/SRH encapsulation is added by the forwarding path in the data plane that also encapsulates the data packets (when the SRv6 path is present in the data plane). This encoding mode requires the Session-Reflector to process two IP headers and a UDP header to locally punt the test packets from the data plane to the CPU or the slow path.

On the other hand, when the Session-Sender test packets are encoded in Insert-Mode, the test packets are generated with an IPv6/SRH encapsulation. For example, when using explicitly configured SRv6 paths, these paths may not be present in the data plane. This encoding mode requires the Session-Reflector to process fewer headers to locally punt the test packets from the data plane to the CPU or the slow path. In this encoding mode, to ensure that the test packets reach the Session-Reflector, PSP is not supported.

In both encoding modes, the timestamps are collected in the data plane, ensuring that the measured delay values are similar.

A Segment List of an SRv6 Policy optionally contains the node SID of the SRv6 Policy endpoint as the ultimate SID. Similarly, the L3/L2 service steered over the SRv6 Policy also ensures that the traffic reaches the endpoint of the SRv6 Policy. Thus, there are two incoming SRv6 SIDs for the Session-Reflector in the packet: the node SID for the endpoint and the SID for the L3/L2 service. As an optimization to avoid processing additional SIDs, the Session-Sender excludes the node SID of the endpoint when carrying an L3/L2 service SID in the packet’s Segment List.

The SRv6 network programming procedures are described in [RFC8986]. The procedure defined for Upper-Layer (UL) Header processing for SRv6 End SIDs in Section 4.1.1 of [RFC8986] is used to process the UDP header in the received Session-Sender test packets on the Session-Reflector.

### 4.3.1. Session-Sender Test Packet for SRv6 Paths

An SRv6 Policy Candidate-Path contains one or more Segment Lists [RFC9256]. For delay measurement of an SRv6 Policy, the Session-Sender test packets are transmitted for every Segment List of the Candidate-Path of the SRv6 Policy by creating a separate STAMP session for each Segment List.

Each Segment List contains a number of SRv6 SIDs as defined in [RFC8986]. The Session-Sender test packets carry the Segment List in an IPv6 header and an SRv6 Segment Routing Header (SRH) [RFC8754].

The content of a Session-Sender test packet for an SRv6 path using the IPv6/SRH encapsulation of the data traffic transmitted over the path is shown in Figure 4. The IPv6/SRH encapsulation is encoded in Insert-Mode or Encaps-Mode. In Insert-Mode, an SRH is inserted after the IPv6 header of the test packets, as shown in Example 1 of Figure 4. In Encaps-Mode, the test packets are encapsulated in an outer IPv6 header with an SRH, as shown in Example 2 of Figure 4.

```
+-----------------------------------------------------------------+
| IPv6 Header                                                     |
.  Source IP Address = Session-Sender IPv6 Address                .
.  Destination IP Address = Segment List[Segments Left]           .
.  Next-Header = 43 (IPv6-Route)                                  .
.                                                                 .
+-----------------------------------------------------------------+
| Routing Type = 4 (SRH)                                          |
.  Segment List[0] = Session-Reflector IPv6 Address or            .
.                    Last Segment of Segment List or              .
.                    Optional PSID                                .
.  <Remained Segment List of Forward Path>                        .
.  Next-Header = 17 (UDP)                                         .
.                                                                 .
+-----------------------------------------------------------------+
| UDP Header and Payload as shown in Figure 3                     |
.                                                                 .
+-----------------------------------------------------------------+
```

Example 1: Encapsulation Using Insert-Mode Encoding

```
+-----------------------------------------------------------------+
| IPv6 Header                                                     |
.  Source IP Address = Session-Sender IPv6 Address                .
.  Destination IP Address = Segment List[Segments Left]           .
.  Next-Header = 43 (IPv6-Route)                                  .
.                                                                 .
+-----------------------------------------------------------------+
| Routing Type = 4 (SRH)                                          |
.  Segment List[0] = Session-Reflector IPv6 Address or            .
.                    Last Segment of Segment List or              .
.                    Optional PSID                                .
.  <Remained Segment List of Forward Path>                        .
.  Next-Header = 41 (IPv6) or 4 (IPv4)                            .
.                                                                 .
+-----------------------------------------------------------------+
| IP Header, UDP Header and Payload as shown in Figure 3          |
.                                                                 .
+-----------------------------------------------------------------+
```

Example 2: Encapsulation Using Encaps-Mode Encoding

Figure 4: Content of Session-Sender Test Packet for SRv6 Path

In the outer IPv6/SRH header, the head-end node address of the SRv6 Policy is used as the Source Address, and the next Segment in the Segment List is used as the Destination Address.  When the Segment List of the Candidate-Path of the SRv6 Policy is empty, the endpoint address of the SRv6 Policy is used as the Destination Address.

In Encaps-Mode for IPv6, an inner IPv6 header is added and contains the endpoint address of the SRv6 Policy as the Destination Address and the head-end node address of the SRv6 Policy as the Source Address.  In the case of an SRv6 Policy with Color-Only Destination Steering, where the endpoint is an unspecified address (the null endpoint :: for IPv6 with all bits set to 0), as defined in Section 8.8.1 of [RFC9256], an IPv6 address from the Dummy IPv6 Prefix address block 100:0:0:1::/64 is used as the Destination Address in the inner IPv6 header of the Session-Sender test packets, instead of using the Session-Reflector Address.  In this case, the Session-Sender ensures that the Session-Sender test packets using the Segment List reach the Session-Reflector at the SRv6 Policy endpoint (for example, by adding the Prefix SID or the IPv6 address of the SRv6 Policy endpoint to the Segment List).  In addition, Session-Sender test packets carry "Destination Node IPv4 or IPv6 Address" STAMP TLV as defined in [RFC9503] to identify the intended Session-Reflector IPv6 address.

In the case of Penultimate Segment Popping (PSP), the IPv6/SRH encapsulation is removed by the penultimate node.  In Insert-Mode, the Session-Sender ensures that the Session-Sender test packets using the Segment List reach the Session-Reflector at the SRv6 Policy endpoint (for example, by adding the Prefix SID or the IPv6 address of the SRv6 Policy endpoint to the Segment List).

The Path Segment Identifier (PSID) [I-D.ietf-spring-srv6-path-segment] of the SRv6 Policy (for the Segment List or for the Candidate-Path) is added to the Segment List of the STAMP test packets when the egress node supports PSID processing.

Each IGP Flex-Algo path in SRv6 networks [RFC9350] has Prefix SIDs advertised by the nodes.  For delay measurement of SRv6 IGP Flex-Algo paths, the Session-Sender test packets carry the SRv6 Flex-Algo Prefix SIDs of the Session-Sender and Session-Reflector as the Source Address and Destination Address in the IPv6 header, respectively, for that SRv6 IGP Flex-Algo path under measurement.

Similarly, each IGP best path in SRv6 networks [RFC9350] has Prefix SIDs advertised by the nodes. For delay measurement of SRv6 IGP best paths, the Session-Sender test packets carry the SRv6 Prefix SIDs of the Session-Sender and Session-Reflector as the Source Address and Destination Address in the IPv6 header, respectively, for that SRv6 best path under measurement.

### 4.3.2. Session-Sender Test Packet for Layer-3 Services over SRv6 Path

For delay measurement of the L3 service over an SRv6 path, the IPv6/SRH encapsulation of the data packets transmitted over the L3 service, including the L3VPN SRv6 SID instantiated on the Session-Reflector (for example, the End.DT6 SID instance, the End.DT4 SID instance, or the End.DT46 instance, as defined in [RFC8986]), is used to encapsulate the Session-Sender test packets, as shown in Figure 5 for both encoding modes: Insert-Mode and Encaps-Mode.

```
+-----------------------------------------------------------------+
| IPv6 Header                                                     |
.  Source IP Address = Session-Sender IPv6 Address                .
.  Destination IP Address = Segment List[Segments Left]           .
.  Next-Header = 43 (IPv6-Route)                                  .
.                                                                 .
+-----------------------------------------------------------------+
| Routing Type = 4 (SRH)                                          |
.  Segment List[0] = End.DT6/End.DT46 SID                         .
.  <Remained Segment List of Forward Path>                        .
.  Next-Header = 17 (UDP)                                         .
.                                                                 .
+-----------------------------------------------------------------+
| UDP Header and Payload as shown in Figure 3                     |
.                                                                 .
+-----------------------------------------------------------------+
```

Example 1: Encapsulation Using Insert-Mode Encoding

```
+-----------------------------------------------------------------+
| IPv6 Header                                                     |
.  Source IP Address = Session-Sender IPv6 Address                .
.  Destination IP Address = Segment List[Segments Left]           .
.  Next-Header = 43 (IPv6-Route)                                  .
.                                                                 .
+-----------------------------------------------------------------+
| Routing Type = 4 (SRH)                                          |
.  Segment List[0] = End.DT4/End.DT46 SID                         .
.  <Remained Segment List of Forward Path>                        .
.  Next-Header = 4 (IPv4)                                         .
.                                                                 .
```

```
+-------------------------------------------------------------+
| IPv4 Header as shown in Figure 3                            |
.  Destination IPv4 Address in L3VPN table                    .
.  Source IPv4 Address in L3VPN table (reverse direction)     .
.                                                             .
+-------------------------------------------------------------+
| UDP Header and Payload as shown in Figure 3                 |
.                                                             .
+-------------------------------------------------------------+

  Example 2: Encapsulation Using Encaps-Mode Encoding for IPv4

+-------------------------------------------------------------+
| IPv6 Header                                                 |
.  Source IP Address = Session-Sender IPv6 Address            .
.  Destination IP Address = Segment List[Segments Left]       .
.  Next-Header = 43 (IPv6-Route)                              .
.                                                             .
+-------------------------------------------------------------+
| Routing Type = 4 (SRH)                                      |
.  Segment List[0] = End.DT6/End.DT46 SID                     .
.  <Remained Segment List of Forward Path>                    .
.  Next-Header = 41 (IPv6)                                    .
.                                                             .
+-------------------------------------------------------------+
| IPv6 Header as shown in Figure 3                            |
.  Destination IPv6 Address in L3VPN table                    .
.  Source IPv6 Address in L3VPN table (reverse direction)     .
.                                                             .
+-------------------------------------------------------------+
| UDP Header and Payload as shown in Figure 3                 |
.                                                             .
+-------------------------------------------------------------+

  Example 3: Encapsulation Using Encaps-Mode Encoding for IPv6
```

Figure 5: Content of Session-Sender Test Packet for L3 Service over SRv6 Path

In Insert-Mode, an SRH is inserted after the IPv6 header of the STAMP test packets, as shown in Example 1 of Figure 5.

In Encaps-Mode, the STAMP test packets are encapsulated in an outer IPv6 header with an SRH, as shown in Examples 2 and 3 of Figure 5.

In both modes, the Session-Sender address is used as the Source Address, and the Session-Reflector address is used as the Destination Address in the outer IPv6 header.

In Encaps-Mode, an inner IP header is added to the Session-Sender test packets after the outer IPv6/SRH encapsulation.

The IPv6 Destination Address added in the inner IPv6 header MUST be reachable via the IPv6 table lookup associated with the L3VPN SRv6 SID added.  Similarly, the IPv4 Destination Address added in the inner IPv4 header MUST be reachable via the IPv4 table lookup associated with the L3VPN SRv6 SID that was added.

The IPv6 Source Address added in the inner IPv6 header MUST be reachable via the IPv6 table lookup for the L3 service in the reverse direction to return the reply test packets over that L3 service. Similarly, the IPv4 Source Address added in the inner IPv4 header MUST be reachable via the IPv4 table lookup for the L3 service in the reverse direction.

#### 4.3.3.  Session-Sender Test Packet for Layer-2 Services over SRv6 Path

For delay measurement of the L2 service over an SRv6 path, the IPv6/SRH encapsulation of the data packets transmitted over the L2 service, including the L2VPN SRv6 SID instantiated on the Session-Reflector (for example, the End.DT2U SID instance as defined in [RFC8986]), is used to encapsulate the Session-Sender test packets, as shown in Figure 6 for both encoding modes: Insert-Mode and Encaps-Mode.

```
    +-----------------------------------------------------------------+
    | IPv6 Header                                                     |
    .  Source IP Address = Session-Sender IPv6 Address                .
    .  Destination IP Address = Segment List[Segments Left]           .
    .  Next-Header = 43 (IPv6-Route)                                  .
    .                                                                 .
    +-----------------------------------------------------------------+
    | Routing Type = 4 (SRH)                                          |
    .  Segment List[0] = End.DT2U SID                                 .
    .  <Remained Segment List of Forward Path>                        .
    .  Next-Header = 17 (UDP)                                         .
    .                                                                 .
    +-----------------------------------------------------------------+
    | UDP Header and Payload as shown in Figure 3                     |
    .                                                                 .
    +-----------------------------------------------------------------+
```

Example 1: Encapsulation Using Insert-Mode Encoding

```
    +-----------------------------------------------------------------+
    | IPv6 Header                                                     |
    .  Source IP Address = Session-Sender IPv6 Address                .
    .  Destination IP Address = Segment List[Segments Left]           .
    .  Next-Header = 43 (IPv6-Route)                                  .
    .                                                                 .
    +-----------------------------------------------------------------+
    | Routing Type = 4 (SRH)                                          |
    .  Segment List[0] = End.DT2U SID                                 .
    .  <Remained Segment List of Forward Path>                        .
    .  Next-Header = 41 (IPv6)                                        .
    .                                                                 .
    +-----------------------------------------------------------------+
    | IPv6 Header as shown in Figure 3                                |
    .  Hop Limit = 1                                                  .
    .                                                                 .
    +-----------------------------------------------------------------+
    | UDP Header and Payload as shown in Figure 3                     |
    .                                                                 .
    +-----------------------------------------------------------------+
```

Example 2: Encapsulation Using Encaps-Mode Encoding

Figure 6: Content of Session-Sender Test Packet for L2 Service over SRv6 Path

In both encoding modes, the Session-Sender address is used as the Source Address, and the Session-Reflector address is used as the Destination Address in the outer IPv6 header.

In Insert-Mode, an SRH is inserted after the IPv6 header of the STAMP test packets, as shown in Example 1 of Figure 6.

In Encaps-Mode, in addition to the outer IPv6/SRH encapsulation, an inner IPv6 header is added, as shown in Example 2 of Figure 6, with a Hop Limit value of 1 to punt the Session-Sender test packets from the data plane to the CPU or the slow path on the Session-Reflector for STAMP processing.  The inner IPv6 header contains the Session-Sender address as the Source Address and the Session-Reflector address as the Destination Address.

## 4.4.  Session-Reflector Test Packet

In two-way measurement mode, the Session-Reflector test packets are transmitted on the same link or the same SRv6 path (i.e., the same set of links and nodes) in the reverse direction to the Session-Sender to perform accurate two-way delay measurement.

The Session-Reflector decapsulates the IPv6/SRH header, if present, from the received Session-Sender test packets.  The Session-Reflector test packet is generated using the information from the received IP/UDP header of the Session-Sender test packet, as shown in Figure 7.

```
+-----------------------------------------------------------------+
| IP Header                                                       |
.  Source IP Address                                              .
.     = Session-Reflector IP Address                              .
.  Destination IP Address                                         .
.     = Source IP Address from Session-Sender Test Packet         .
.  IPv4 Protocol or IPv6 Next-header = 17 (UDP)                   .
.                                                                 .
+-----------------------------------------------------------------+
| UDP Header                                                      |
.  Source Port = Chosen by Session-Reflector                      .
.  Destination Port                                               .
.     = Source Port from Session-Sender Test Packet               .
.                                                                 .
+-----------------------------------------------------------------+
| Payload = Test Packet as specified in Section 3 of RFC 8972    |
.           in Figures 2 and 4                                    .
.                                                                 .
+-----------------------------------------------------------------+
```

Figure 7: Content of Session-Reflector Test Packet

The payload contains the Session-Reflector test packet defined in Section 3 of [RFC8972].

In the case of links, the IPv6/SRH is not present in the received Session-Sender test packet.  The Session-Sender sets the "Reply Requested on the Same Link" flag in the Control Code Sub-TLV in the Return Path TLV defined in [RFC9503] to request the Session-Reflector to transmit the reply test packet on the same link in the reverse direction.

For SRv6 paths, the Session-Sender uses the Segment List sub-TLV in the Return Path TLV defined in [RFC9503] to request that the Session-Reflector transmit the reply test packet on a specific SRv6 return path.  Examples of specific SRv6 return paths include: the reverse SRv6 path associated with the forward direction SRv6 path, the Binding SID of the reverse SRv6 Policy, or the SRv6 Prefix SID of the Session-Sender.

For SRv6 IGP Flex-Algo paths, the Session-Sender uses the Segment List sub-TLV in the Return Path TLV defined in [RFC9503] to request that the Session-Reflector transmit the reply test packet on the same SRv6 IGP Flex-Algo path in the reverse direction.

## 5.  One-Way Measurement Mode

As shown in Figure 8, the reference topology for one-way measurement mode, the STAMP Session-Sender S1 initiates a Session-Sender test packet.  The STAMP Session-Reflector does not transmit reply test packets upon receiving the Session-Sender test packets.

T1 is a transmit timestamp added by node S1, and T2 is a receive timestamp added by node R1.  Timestamps T1 and T2 are used by the Session-Reflector to measure the one-way delay metric as (T2 - T1).

The computation of the one-way delay metric requires the clocks on the Session-Sender and Session-Reflector to be synchronized using either PTPv2 or NTPv4.

```
                    T1                 T2
                   /                     \
          +-------+     Test Packet     +-------+
          |       | - - - - - - - - - ->|       |
          |   S1  |=====================|   R1  |
          |       |                     |       |
          +-------+                     +-------+

    STAMP Session-Sender          STAMP Session-Reflector
```

Figure 8: Reference Topology for One-Way Measurement Mode

## 5.1. STAMP Reference Model Considerations for One-Way Measurement Mode

In one-way measurement mode, for links, SRv6 paths, and L3 and L2 services over the SRv6 paths, the Session-Sender test packets, as defined in Section 4 for STAMP sessions, are transmitted.

The Stateful mode of the Session-Reflector [RFC8762] is used as the Session-Receiver in one-way measurement mode. The SSID field in the received Session-Sender test packets [RFC8972] at the Session-Reflector, along with the local configuration, is used to identify the STAMP sessions that use one-way measurement mode on the Stateful Session-Reflector.

Typically, a different destination UDP port is selected for one-way measurement mode than the one used by the STAMP Session-Reflector for two-way measurement mode. When the same STAMP Session-Reflector UDP port is selected for one-way measurement mode, the Session-Sender requests, in the test packets, that the Session-Reflector not transmit reply test packets. To achieve this, it uses the "No Reply Requested" flag in the Control Code Sub-TLV within the Return Path TLV defined in [RFC9503].

## 6. Loopback Measurement Mode

As shown in Figure 9, the reference topology for loopback measurement mode, the STAMP Session-Sender S1 initiates a Session-Sender test packet to measure the loopback delay of a bidirectional path. At the STAMP Session-Reflector, the received Session-Sender test packets are not punted out of the fast path in the data plane (i.e., to the CPU or the slow path) but are simply forwarded. In other words, the Session-Reflector does not perform STAMP functions or generate Session-Reflector test packets.

```
                       T1
                      /
             +-------+     Test Packet     +-------+
             |       | - - - - - - - - - - |       |
             |   S1  |====================||   R1  |
             |       |<- - - - - - - - - - |       |
             +-------+  Return Test Packet +-------+
                      \
                       T4

       STAMP Session-Sender          STAMP Session-Reflector
                                           (Loopback,
                                            Forward)
```

Figure 9: Reference Topology for Loopback Measurement Mode

The Session-Sender retrieves the timestamp T1 from the received Session-Sender test packet and collects the receive timestamp T4 locally. Both timestamps, T1 and T4, are used to measure the loopback delay metric as (T4 - T1). The loopback delay includes the STAMP test packet processing delay on the Session-Reflector component. The Session-Reflector processing delay component includes only the time required to loop the STAMP test packet from the incoming interface to the outgoing interface in the data plane. The Session-Reflector does not timestamp the test packets and, therefore, does not require timestamping capability.

### 6.1. STAMP Reference Model Considerations for Loopback Measurement Mode

The Session-Sender test packets are encapsulated with the forward direction SRv6 path and transmitted to the Session-Reflector, as defined in Section 4 for STAMP sessions. An IP header is added for the return path in the Session-Sender test packets, setting the Destination Address equal to the Session-Sender address, as shown in Figure 10, to return the test packets to the Session-Sender.

```
+----------------------------------------------------------------+
| IP Header (Return Path)                                        |
.  Source IP Address = Session-Sender IP Address                 .
.  Destination IP Address = Session-Sender IP Address            .
.  IPv4 Protocol or IPv6 Next-header = 17 (UDP)                  .
.                                                                .
+----------------------------------------------------------------+
| UDP Header                                                     |
.  Source Port = Chosen by Session-Sender                        .
.  Destination Port = Source Port                                .
.                                                                .
+----------------------------------------------------------------+
| Payload = Test Packet as specified in Section 3 of RFC 8972   |
.           in Figures 1 and 3                                   .
.                                                                .
+----------------------------------------------------------------+
```

Figure 10: Content of Session-Sender Return Test Packet in Loopback Measurement Mode

The Session-Reflector does not perform the STAMP process, as the loopback function simply processes the encapsulation, including the IPv6/SRH headers (but does not process the UDP header) to forward the received Session-Sender test packet to the Session-Sender without STAMP modifications, as defined in [RFC8762].

The SSID field in the received Session-Sender test packets [RFC8972] at the Session-Sender, along with the local configuration, is used to identify the STAMP sessions that use loopback measurement mode.

The Session-Sender sets the destination UDP port to the UDP port it uses to receive the return Session-Reflector test packets (other than destination UDP port 862, which is used by the Session-Reflector). The same UDP port is used as both the destination and source UDP port in the Session-Sender test packets, as shown in Figure 10.

At the Session-Sender, the 'Session-Sender Sequence Number,' 'Session-Sender Timestamp,' 'Session-Sender Error Estimate,' and 'Session-Sender TTL' fields are set to zero in the transmitted Session-Sender test packets and are ignored in the received test packets.

## 6.2. Loopback Measurement Mode for Links

The Session-Sender test packets in loopback measurement mode for Ethernet links are transmitted with a Layer-2 header for the forward direction path. The Layer-2 header contains the link MAC address on the Session-Reflector as the Destination Address and the link MAC address on the Session-Sender as the Source MAC address, as shown in Figure 11.

```
+-----------------------------------------------------------------+
| L2 MAC Header (Forward Path)                                    |
.  Source Address = Link MAC Address on Session-Sender            .
.  Destination Address = Link MAC Address on Session-Reflector    .
.  Ether-Type = 0x0800 (IPv4) Or 0x86DD (IPv6)                    .
.                                                                 .
+-----------------------------------------------------------------+
| Test Packet as shown in Figure 10 (Return Path)                 |
.                                                                 .
+-----------------------------------------------------------------+
```

Figure 11: Content of Session-Sender Test Packet in Loopback Measurement Mode for Ethernet Link

The IP header for the return path of the Session-Sender test packets is also added, with the Source and Destination Addresses set equal to the link address on the Session-Sender to return the test packet to the Session-Sender.

The Session-Reflector decapsulates the Layer-2 header and forwards the test packet using the IP header to the Session-Sender.

## 6.3. Loopback Measurement Mode for SRv6 Paths

In loopback measurement mode for SRv6 paths, the Session-Sender test packet carries either the Segment List of the forward direction path only (using Encaps-Mode encoding), or both the forward direction and return paths in IPv6/SRH (using Insert-Mode encoding), as shown in Figure 12.

```
+-----------------------------------------------------------------+
| IPv6 Header                                                     |
.  Source IP Address = Session-Sender IPv6 Address                .
.  Destination IP Address = Segment List[Segments Left]           .
.  Next-Header = 43 (IPv6-Route)                                  .
.                                                                 .
+-----------------------------------------------------------------+
| Routing Type = 4 (SRH)                                          |
.  Segment List[0] = Session-Sender IPv6 Address or               .
.                    Last Segment of Segment List of Return Path.
.                    or Optional PSID of Return Path              .
.  <Remained Segment List for Return Path>                        .
.  <Optional PSID of Forward Path>                                .
.  <Remained Segment List for Forward Path>                       .
.  Next-Header = 17 (UDP)                                         .
.                                                                 .
+-----------------------------------------------------------------+
| UDP Header and Payload as shown in Figure 10                    |
.                                                                 .
+-----------------------------------------------------------------+

     Example 1: Encapsulation Using Insert-Mode Encoding
                with SRv6 Return Path

+-----------------------------------------------------------------+
| IPv6 Header                                                     |
.  Source IP Address = Session-Sender IPv6 Address                .
.  Destination IP Address = Segment List[Segments Left]           .
.  Next-Header = 43 (IPv6-Route)                                  .
.                                                                 .
+-----------------------------------------------------------------+
| Routing Type = 4 (SRH)                                          |
.  Segment List[0] = Session-Reflector IPv6 Address or            .
.                    Last Segment of Segment List or              .
.                    Optional PSID of Forward Path                .
.  <Remained Segment List of Forward Path>                        .
.  Next-Header = 41 (IPv6) or 4 (IPv4)                            .
.                                                                 .
+-----------------------------------------------------------------+
| IP Header as shown in Figure 10 (Return Path)                   |
```

```
.                                                               .
+---------------------------------------------------------------+
| UDP Header and Payload as shown in Figure 10                  |
.                                                               .
+---------------------------------------------------------------+
```

Example 2: Encapsulation Using Encaps-Mode Encoding with IP Return Path

Figure 12: Content of Session-Sender Test Packet in Loopback Measurement Mode for SRv6 Path

The Session-Sender ensures that the Session-Sender test packets using the Segment List reach the SRv6 Policy endpoint, for example, by adding the Prefix SID or IPv6 address of the SRv6 Policy endpoint to the Segment List, in both encoding modes.

### 6.3.1. SRv6 Return Path

For the SRv6 return path, the Session-Sender test packets are encoded in Insert-Mode, as shown in Example 1 of Figure 12.

The Session-Sender test packets, in the SRv6 Segment List, carry the return path in addition to the forward direction path. For example, they may carry the Segment List of the associated reverse Candidate-Path, the Binding SID of the reverse SRv6 Policy, or the SRv6 Prefix SID of the Session-Sender. The Binding SID of the reverse SRv6 Policy can be configured on the Session-Sender using an SDN controller, for example.

For SRv6 IGP Flex-Algo paths, the Session-Sender test packets carry the SRv6 Prefix SID of the Session-Sender on the same IGP Flex-Algo path in the reverse direction.

The PSID is added to the Segment List of the Session-Sender test packets for the SRv6 return path when the head-end node supports PSID allocation.

Encaps-Mode using an SRv6 return path does not preclude carrying an inner IP header of the IP return path.

### 6.3.2. IP Return Path

For the IP return path, the Session-Sender test packets are encoded in Encaps-Mode, as shown in Example 2 of Figure 12.

The Session-Sender test packets carry the Segment List of the SRv6 forward direction path only.

An inner IP header for the return path is added to the Session-Sender test packets, with the Destination Address set to the Session-Sender address to return the test packet to the Session-Sender.

The Session-Reflector decapsulates the IPv6/SRH headers and forwards the test packet using the inner IP header for the return path.

The optional PSID added to the Session-Sender test packet is for the SRv6 forward direction path and is allocated by the Session-Reflector.

### 6.4. Loopback Measurement Mode for Layer-3 Services over SRv6 Path

In loopback measurement mode for the L3 service over an SRv6 path, the IPv6/SRH encapsulation of the data packets transmitted over the L3 service, including the L3VPN SRv6 SID (e.g., the End.DT6 SID instance, the End.DT4 SID instance, etc., as defined in [RFC8986]), is used to encapsulate the Session-Sender test packets, as shown in Figure 13.

```
+-----------------------------------------------------------------+
| IPv6 Header                                                     |
.  Source IP Address = Session-Sender IPv6 Address                .
.  Destination IP Address = Segment List[Segments Left]           .
.  Next-Header = 43 (IPv6-Route)                                  .
.                                                                 .
+-----------------------------------------------------------------+
| Routing Type = 4 (SRH)                                          |
.  Segment List[0] = End.DT4/DT6/DT46 SID of Return Path          .
.  <Remained Segment List of Return Path>                         .
.  <Remained Segment List of Forward Path>                        .
.  Next-Header = 17 (UDP)                                         .
.                                                                 .
+-----------------------------------------------------------------+
| UDP Header and Payload as shown in Figure 10                    |
.                                                                 .
+-----------------------------------------------------------------+
```

Example 1: Encapsulation Using Insert-Mode Encoding
with SRv6 Return Path

```
+-----------------------------------------------------------------+
| IPv6 Header                                                     |
.  Source IP Address = Session-Sender IPv6 Address                .
.  Destination IP Address = Segment List[Segments Left]           .
.  Next-Header = 43 (IPv6-Route)                                  .
.                                                                 .
+-----------------------------------------------------------------+
```

```
    | Routing Type = 4 (SRH)                                          |
    .  Segment List[0] = End.DT4/DT46 SID of Forward Path             .
    .  <Remained Segment List of Forward Path>                        .
    .  Next-Header = 4 (IPv4)                                         .
    .                                                                 .
    +-----------------------------------------------------------------+
    | IPv4 Header as shown in Figure 10 (Return Path)                 |
    .      Destination IPv4 Address in L3VPN table                    .
    +-----------------------------------------------------------------+
    | UDP Header and Payload as shown in Figure 10                    |
    .                                                                 .
    +-----------------------------------------------------------------+

        Example 2: Encapsulation Using Encaps-Mode Encoding
                   with IPv4 Return Path

    +-----------------------------------------------------------------+
    | IPv6 Header                                                     |
    .  Source IP Address = Session-Sender IPv6 Address                .
    .  Destination IP Address = Segment List[Segments Left]           .
    .  Next-Header = 43 (IPv6-Route)                                  .
    .                                                                 .
    +-----------------------------------------------------------------+
    | Routing Type = 4 (SRH)                                          |
    .  Segment List[0] = End.DT6/DT46 SID of Forward Path             .
    .  <Remained Segment List of Forward Path>                        .
    .  Next-Header = 41 (IPv6)                                        .
    .                                                                 .
    +-----------------------------------------------------------------+
    | IPv6 Header as shown in Figure 10 (Return Path)                 |
    .      Destination IPv6 Address in L3VPN table                    .
    +-----------------------------------------------------------------+
    | UDP Header and Payload as shown in Figure 10                    |
    .                                                                 .
    +-----------------------------------------------------------------+

        Example 3: Encapsulation Using Encaps-Mode Encoding
                   with IPv6 Return Path
```

Figure 13: Content of Session-Sender Test Packet in Loopback Measurement Mode for L3 Service over SRv6 Path

### 6.4.1. SRv6 Return Path

For the SRv6 return path, the Session-Sender test packets are encoded in Insert-Mode, as shown in Example 1 of Figure 13.

The SRv6 Segment List, except for the L3VPN SRv6 SID instantiated on the Session-Reflector for the forward direction L3 service, is added to the IPv6/SRH encapsulation of the Session-Sender test packet. In addition, the SRv6 Segment List, including the L3VPN SRv6 SID instantiated on the Session-Sender for the reverse direction L3 service, is also added to the IPv6/SRH encapsulation to return the test packet to the Session-Sender from the Session-Reflector.

Encaps-Mode using an SRv6 return path does not preclude carrying an inner IP header of the IP return path.

#### 6.4.2. IP Return Path

For the IP return path, the Session-Sender test packets are encoded in Encaps-Mode, as shown in Examples 2 and 3 of Figure 13.

The SRv6 Segment List, including the L3VPN SRv6 SID instantiated on the Session-Reflector for the forward direction L3 service, is added to the IPv6/SRH to encapsulate the Session-Sender test packets sent to the Session-Reflector.

An inner IP header for the return path is also added to the Session-Sender test packets, with the Destination Address set to the Session-Sender address to forward the test packet to the Session-Sender from the Session-Reflector. In this case, the Destination Address added in the inner IP header for the return path MUST be reachable via the IPv4 or IPv6 table lookup associated with the L3VPN SRv6 SID on the Session-Reflector.

The Session-Reflector decapsulates the IPv6/SRH and forwards the Session-Sender test packet using the inner IP header, after adding IPv6/SRH encapsulation for the reverse direction L3 service.

### 6.5. Loopback Measurement Mode for Layer-2 Services over SRv6 Path

In loopback measurement mode for the L2 service over an SRv6 path, the IPv6/SRH encapsulation of the data packets transmitted over the L2 service, including the L2VPN SRv6 SID (e.g., the End.DT2U SID instance, as defined in [RFC8986]), is used to encapsulate the Session-Sender test packets, as shown in Figure 14.

```
+-----------------------------------------------------------------+
| IPv6 Header                                                     |
.  Source IP Address = Session-Sender IPv6 Address                .
.  Destination IP Address = Segment List[Segments Left]           .
.  Next-Header = 43 (IPv6-Route)                                  .
.                                                                 .
+-----------------------------------------------------------------+
| Routing Type = 4 (SRH)                                          |
.  Segment List[0] = End.DT2U SID of Return Path                  .
.  <Remained Segment List of Return Path>                         .
.  <Remained Segment List of Forward Path>                        .
.  Next-Header = 17 (UDP)                                         .
.                                                                 .
+-----------------------------------------------------------------+
| UDP Header and Payload as shown in Figure 10                    |
.                                                                 .
+-----------------------------------------------------------------+
```

Encapsulation Using Insert-Mode Encoding with SRv6 Return Path

Figure 14: Content of Session-Sender Test Packet in Loopback Mode for L2 Service over SRv6 Path

#### 6.5.1. SRv6 Return Path

For the SRv6 return path, the Session-Sender test packets are encoded in Insert-Mode, as shown in Figure 14.

The SRv6 Segment List, except for the L2VPN SRv6 SID instantiated on the Session-Reflector for the forward direction L2 service, is added to the IPv6/SRH encapsulation of the Session-Sender test packet. In addition, the SRv6 Segment List, including the L2VPN SRv6 SID instantiated on the Session-Sender for the reverse direction L2 service, is also added to the IPv6/SRH encapsulation to return the test packet to the Session-Sender from the Session-Reflector.

#### 6.5.2. IP Return Path

The STAMP test packets that do not use the SRv6 return path are not supported.

## 7. Loopback Measurement Mode with Timestamp and Forward

As shown in Figure 15, the reference topology for "loopback measurement mode with timestamp and forward", the STAMP Session-Sender S1 initiates a Session-Sender test packet in loopback measurement mode. The "timestamp and forward" mechanism is used to optimize the "operations of punting the test packet and generating the return test packet" on the STAMP Session-Reflector, as timestamping is implemented in the fast path in the data plane. This helps achieve a higher number of STAMP sessions and faster measurement intervals.

```
                         T1                 T2
                        /                    \
               +-------+     Test Packet     +-------+
               |       | - - - - - - - - - - |       |
               |   S1  |====================||   R1  |
               |       |<- - - - - - - - - - |       |
               +-------+  Return Test Packet +-------+
                        \
                         T4

         STAMP Session-Sender          STAMP Session-Reflector
                                             (Loopback,
                                              Timestamp and Forward)
```

Figure 15: Reference Topology for Loopback Measurement Mode with Timestamp and Forward

The Session-Sender retrieves the timestamps T1 and T2 from the received Session-Sender test packet and collects the receive timestamp T4 locally. Timestamps T1 and T2 are used by the Session-Sender to measure the one-way delay metric as (T2 - T1). Timestamps T1 and T4 are used by the Session-Sender to measure the loopback delay metric as (T4 - T1).

The Session-Sender adds the transmit timestamp (T1) to the payload of the Session-Sender test packet. The Session-Reflector adds the receive timestamp (T2) to the payload of the received test packet in the fast path in the data plane, without punting the test packet (e.g., to the CPU or the slow path) for STAMP packet processing.

### 7.1. Loopback Measurement Mode with Timestamp and Forward Endpoint Behaviour for SRv6 Data Plane

[RFC8986] defines SRv6 Endpoint Behaviours for SRv6 nodes. A new SRv6 Endpoint Behaviour, the "Timestamp and Forward (End.TSF)" (value TBA1), is defined for STAMP test packets.

In the Session-Sender test packets for SRv6 paths, the "Timestamp and Forward" Endpoint Behaviour (End.TSF) is carried with the target Segment Identifier (SID) in the SRH [RFC8754], as shown in Figure 16, for both Insert-Mode and Encaps-Mode encoding, to collect timestamps in the "Receive Timestamp" field in the payload of the test packet from the Session-Reflector.

```
    +-----------------------------------------------------------------+
    | IPv6 Header                                                     |
    .  Source IP Address = Session-Sender IPv6 Address                .
    .  Destination IP Address = Segment List[Segments Left]           .
    .  Next-Header = 43 (IPv6-Route)                                  .
    .                                                                 .
    +-----------------------------------------------------------------+
    | Routing Type = 4 (SRH)                                          |
    .  &lt;Segment List for Return Path&gt;                                 .
    .  &lt;Segment List for Forward Path including End.TSF SID&gt;          .
    .  Next-Header = 17 (UDP)                                         .
    .                                                                 .
    +-----------------------------------------------------------------+
    | UDP Header and Payload as shown in Figure 10                    |
    .                                                                 .
    +-----------------------------------------------------------------+

        Example 1: Encapsulation Using Insert-Mode Encoding
                   with SRv6 Return Path

    +-----------------------------------------------------------------+
    | IPv6 Header                                                     |
    .  Source IP Address = Session-Sender IPv6 Address                .
    .  Destination IP Address = Segment List[Segments Left]           .
    .  Next-Header = 43 (IPv6-Route)                                  .
    .                                                                 .
    +-----------------------------------------------------------------+
    | Routing Type = 4 (SRH)                                          |
    .  Segment List[0] = End.TSF SID                                  .
    .  <Remained Segment List of Forward Path>                        .
    .  Next-Header = 41 (IPv6) or 4 (IPv4)                            .
    .                                                                 .
    +-----------------------------------------------------------------+
    | IP Header as shown in Figure 10 (Return Path)                   |
    .                                                                 .
    +-----------------------------------------------------------------+
    | UDP Header and Payload as shown in Figure 10                    |
    .                                                                 .
    +-----------------------------------------------------------------+

        Example 2: Encapsulation Using Encaps-Mode Encoding
                   with IP Return Path
```

Figure 16: Content of Session-Sender Test Packet in Loopback Measurement Mode with End.TSF for SRv6 Paths

The Session-Sender test packets are encoded in Insert-Mode for the SRv6 return path and in Encaps-Mode for the IP return path, as defined in the loopback measurement mode for SRv6 paths in this document.

When a Session-Reflector receives a test packet with the Timestamp and Forward (End.TSF) Endpoint Behaviour for the target SID, which is local, it timestamps the test packet at a specific offset and then forwards the test packet as defined in the loopback measurement mode for SRv6 paths.

#### 7.1.1. Timestamp and Forward Endpoint Behaviour Assignment and Node Capability

A new SRv6 endpoint behaviour, "Timestamp and Forward (End.TSF)", bound to SRv6 SID and instantiated on the Session-Reflector node, with value TBA1 (to be assigned by IANA) is defined in this document.

The timestamp format (e.g., 64-bit PTPv2 or NTPv4), to be added to the Session-Sender test packet payload, is locally configured for the End.TSF endpoint behaviour. The offset in the Session-Sender test packet payload (e.g., STAMP test packet in Figure 5 of [RFC8762] with an offset of 16 bytes for Receive Timestamp) is also locally configured for the End.TSF endpoint behaviour.

The Session-Sender needs to know if the Session-Reflector is capable of processing the "Timestamp and Forward" Endpoint Behaviour to avoid dropping the test packets. The signaling extension for this capability exchange or its configuration through local settings is outside the scope of this document.

## 8. Packet Loss Measurement in SRv6 Networks

The procedure described for two-way measurement mode allows for round-trip, near-end (forward direction), and far-end (backward direction) inferred packet loss measurement. However, this provides only an approximate view of the data packet loss.

The loopback measurement mode and loopback measurement mode with "timestamp and forward", defined in this document, allow only round-trip packet loss measurement.

Note that the packet loss measurement does not require the clocks on the Session-Sender and Session-Reflector to be synchronized using either PTPv2 or NTPv4.

## 9. Direct Measurement in SRv6 Networks

The STAMP "Direct Measurement" TLV (Type 5), defined in [RFC8972], is used for data packet loss measurement. The STAMP test packets with this TLV are transmitted using the procedure described for two-way measurement mode using STAMP test packets and collect the Session-Sender transmit counters and Session-Reflector receive and transmit counters of the data packet flows for direct measurement.

The PSID carried in the data packets is used to measure received data packets (for the receive traffic counter) on the associated SRv6 path on the Session-Reflector.

In the case of L3 and L2 services over the SRv6 paths, the associated SRv6 service SIDs are used to measure received data packets (for the receive traffic counters) on the Session-Reflector.

In loopback measurement mode and loopback measurement mode with "timestamp and forward", defined in this document, direct measurement is not applicable.

## 10. ECMP Measurement in SRv6 Networks

The Segment List of an SRv6 path can have ECMP paths between the source and transit nodes, between transit nodes, and between transit and destination nodes. The usage of a node SID [RFC8402] by the Segment List of an SRv6 path can result in ECMP paths. In addition, the usage of an Anycast SID [RFC8402] by the Segment List of an SRv6 path can result in ECMP paths via transit nodes that are part of that anycast group. The STAMP test packets are transmitted to traverse different ECMP paths to measure the delay of each ECMP path of a Segment List.

As specified in [RFC6437], different values of the Flow Label field in the outer IPv6 header of the Session-Sender and Session-Reflector test packets are used to traverse different IPv6 ECMP paths for delay measurement.

The considerations for loss measurement for different ECMP paths of an SRv6 path are outside the scope of this document.

## 11. STAMP Session State

The threshold-based notification for the delay and packet loss metrics is not generated if the delay and packet loss metrics do not change significantly. For unambiguous monitoring, the controller needs to distinguish whether the STAMP session is active but delay and packet loss metrics do not cross the thresholds, or if the STAMP session has failed and is not transmitting or receiving test packets.

The STAMP session state monitoring allows the node to determine whether the performance measurement test is active, idle, or failed. The STAMP session state is notified as idle when the Session-Sender is not transmitting test packets. The STAMP session state is initially notified as active when the Session-Sender is transmitting test packets and as soon as one or more reply test packets are received at the Session-Sender.

The STAMP session state is notified as failed when N consecutive reply test packets are not received at the Session-Sender after the STAMP session state is notified as active, where N (the consecutive packet loss count) is a locally provisioned value. In this case, the failed state of the STAMP session on the Session-Sender also indicates the connectivity failure of the link, SRv6 path, or L3/L2 service over the SRv6 path, where the STAMP session was active.

## 12. Additional STAMP Test Packet Processing Rules

### 12.1. TTL

The TTL field in the IPv4 headers of the Session-Sender and Session-Reflector test packets is set to 255, as per the Generalized TTL Security Mechanism (GTSM) [RFC5082].

### 12.2. IPv6 Hop Limit

The Hop Limit (HL) field in all IPv6 headers of the Session-Sender and Session-Reflector test packets is set to 255, as per the Generalized TTL Security Mechanism (GTSM) [RFC5082].

### 12.3. Router Alert Option

The Router Alert IP option (RAO) [RFC2113] is not required in the Session-Sender and Session-Reflector test packets to punt the STAMP test packets from the data plane to the CPU or the slow path.

### 12.4. IPv6 Flow Label

The Flow Label field in the IPv6 header of the Session-Sender test packets is set to the value used by the data packets for the IPv6 traffic flow being measured by the Session-Sender.

The Session-Reflector uses the Flow Label value received in the IPv6 header of the Session-Sender test packet for the reply test packet, which can be based on a local policy.

### 12.5. UDP Checksum

For IPv6 STAMP test packets, where the local processor, after adding the timestamp, is not capable of re-computing the UDP checksum or adding a checksum complement [RFC7820], the Session-Sender and Session-Reflector use the procedure defined in [RFC6936] for the UDP checksum (with the value set to 0) for UDP ports used in STAMP sessions, which can be based on a local policy.

## 13. Implementation Status

Editorial note: Please remove this section prior to publication.

### 13.1. Cisco Implementation

The following Cisco routing platforms running the IOS-XR operating system have participated in interoperability testing for one-way, two-way and loopback measurement modes for links and SRv6:

* Cisco 8000 (based on Cisco Silicon One ASIC)

* Cisco ASR9904 with Lightspeed line card and Tomahawk line card

* Cisco NCS5500 (based on Broadcom Jericho1 ASIC)

* Cisco NCS5700 (based on Broadcom Jericho2 ASIC)

### 13.2. Teaparty Implementation

An open-source implementation of the Simple Two-Way Active Measurement Protocol [RFC8762] is available in Teaparty.

https://github.com/cerfcast/teaparty

An implementation of the solution defined in [RFC9503] is available at the following location:

https://github.com/cerfcast/teaparty/commit/393abf9357a6c2439877d9bcf2dc426dd89c7158

The features implemented are:

1.  Destination Node Address TLV.

2.  Return Path TLV.

There is also support for these TLVs in the Wireshark dissector:

https://github.com/cerfcast/teaparty/commit/fb74e2e02396e9bb3ead017e8d9a0c187e3573e2

Contact:

William Hawkins

University of Cincinnati

Email: hawkinsw@obs.cr

## 14.  Operational and Manageability Considerations

The operational considerations described in Section 5 of [RFC8762] and the manageability considerations described in Section 9 of [RFC8402] apply to this specification.

Various statistics for one-way (near-end, far-end), round-trip, and loopback delay metrics (such as, average delay, minimum delay, maximum delay, and delay-variance) as well as for one-way (near-end, far-end) or round-trip packet loss metrics (such as, percentage loss and consecutive packets lost) can be computed using the performance measurement procedures described in this document.  Operator alerts are generated for anomaly detection when delay or loss metrics cross user-configured thresholds.

When STAMP sessions are created for the Segment Lists of the SRv6 Policies, the scalability regarding the number of STAMP sessions needs to be carefully considered.

## 15.  Security Considerations

The security considerations specified in [RFC8762], [RFC8972], and [RFC9503] also apply to the procedures described in this document.

The use of HMAC-SHA-256 in authenticated mode protects the data integrity of the STAMP test packets. The message integrity protection using HMAC, as defined in Section 4.4 of [RFC8762], can be used with the procedures described in this document.

STAMP uses a well-known UDP port number that could become a target of Denial of Service (DoS) attacks or could be used to aid in on-path attacks. Thus, the security considerations and measures to mitigate the risk of such attacks, as documented in Section 6 of [RFC8545], equally apply to the procedures described in this document.

The procedures defined in this document are intended for deployment in a single network administrative domain. As such, the Session-Sender address, Session-Reflector address, and the forward direction and return paths are provisioned by the operator for the STAMP session. It is assumed that the operator has verified the integrity of the forward direction and return paths of the STAMP test packets.

When using the procedures defined in [RFC6936], the security considerations specified in [RFC6936] also apply.

The STAMP test packets for SRv6 can use the HMAC authentication defined for SRH in [RFC8754].

The security considerations specified in [RFC8986] are also applicable to the procedures defined in this document.

## 16. IANA Considerations

This document requests IANA to allocate the following codepoint within the First Come First Served, "SRv6 Endpoint Behaviours" sub-registry, under the top-level "Segment Routing Parameters" registry.

| Value | Hex | Endpoint Behaviour | Reference | Change Controller |
|---|---|---|---|---|
| TBA1 | TBA1-HEX | Timestamp and Forward (TSF) | This document | IETF |

Table 1: SRv6 Endpoint Behaviour

## 17. References

### 17.1. Normative References

[RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, DOI 10.17487/RFC0768, August 1980, <https://www.rfc-editor.org/info/rfc768>.

[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>.

[RFC8762] Mirsky, G., Jun, G., Nydell, H., and R. Foote, "Simple Two-Way Active Measurement Protocol", RFC 8762, DOI 10.17487/RFC8762, March 2020, <https://www.rfc-editor.org/info/rfc8762>.

[RFC8972] Mirsky, G., Min, X., Nydell, H., Foote, R., Masputra, A., and E. Ruffini, "Simple Two-Way Active Measurement Protocol Optional Extensions", RFC 8972, DOI 10.17487/RFC8972, January 2021, <https://www.rfc-editor.org/info/rfc8972>.

[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>.

[RFC9503] Gandhi, R., Ed., Filsfils, C., Chen, M., Janssens, B., and R. Foote, "Simple Two-Way Active Measurement Protocol (STAMP) Extensions for Segment Routing Networks", RFC 9503, DOI 10.17487/RFC9503, October 2023, <https://www.rfc-editor.org/info/rfc9503>.

[RFC9534] Li, Z., Zhou, T., Guo, J., Mirsky, G., and R. Gandhi, "Simple Two-Way Active Measurement Protocol Extensions for Performance Measurement on a Link Aggregation Group", RFC 9534, DOI 10.17487/RFC9534, January 2024, <https://www.rfc-editor.org/info/rfc9534>.

[RFC9800] Cheng, W., Ed., Filsfils, C., Li, Z., Decraene, B., and F. Clad, Ed., "Compressed SRv6 Segment List Encoding", RFC 9800, DOI 10.17487/RFC9800, June 2025, <https://www.rfc-editor.org/info/rfc9800>.

[I-D.ietf-spring-srv6-path-segment]
Li, C., Cheng, W., Zeng, G., Dhody, D., and Y. Zhu, "Path Segment Identifier (PSID) in SRv6 (Segment Routing in IPv6)", Work in Progress, Internet-Draft, draft-ietf-spring-srv6-path-segment-15, 15 March 2026, <https://datatracker.ietf.org/doc/html/draft-ietf-spring-srv6-path-segment-15>.

## 17.2. Informative References

[RFC2113] Katz, D., "IP Router Alert Option", RFC 2113, DOI 10.17487/RFC2113, February 1997, <https://www.rfc-editor.org/info/rfc2113>.

[RFC5082] Gill, V., Heasley, J., Meyer, D., Savola, P., Ed., and C. Pignataro, "The Generalized TTL Security Mechanism (GTSM)", RFC 5082, DOI 10.17487/RFC5082, October 2007, <https://www.rfc-editor.org/info/rfc5082>.

[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/info/rfc5905>.

[RFC6437] Amante, S., Carpenter, B., Jiang, S., and J. Rajahalme, "IPv6 Flow Label Specification", RFC 6437, DOI 10.17487/RFC6437, November 2011, <https://www.rfc-editor.org/info/rfc6437>.

[RFC6936] Fairhurst, G. and M. Westerlund, "Applicability Statement for the Use of IPv6 UDP Datagrams with Zero Checksums", RFC 6936, DOI 10.17487/RFC6936, April 2013, <https://www.rfc-editor.org/info/rfc6936>.

[RFC7404] Behringer, M. and E. Vyncke, "Using Only Link-Local Addressing inside an IPv6 Network", RFC 7404, DOI 10.17487/RFC7404, November 2014, <https://www.rfc-editor.org/info/rfc7404>.

[RFC7820] Mizrahi, T., "UDP Checksum Complement in the One-Way Active Measurement Protocol (OWAMP) and Two-Way Active Measurement Protocol (TWAMP)", RFC 7820, DOI 10.17487/RFC7820, March 2016, <https://www.rfc-editor.org/info/rfc7820>.

[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>.

[RFC8545] Morton, A., Ed. and G. Mirsky, Ed., "Well-Known Port Assignments for the One-Way Active Measurement Protocol (OWAMP) and the Two-Way Active Measurement Protocol (TWAMP)", RFC 8545, DOI 10.17487/RFC8545, March 2019, <https://www.rfc-editor.org/info/rfc8545>.

[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>.

[RFC9256] Filsfils, C., Talaulikar, K., Ed., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", RFC 9256, DOI 10.17487/RFC9256, July 2022, <https://www.rfc-editor.org/info/rfc9256>.

[RFC9350] Psenak, P., Ed., Hegde, S., Filsfils, C., Talaulikar, K., and A. Gulko, "IGP Flexible Algorithm", RFC 9350, DOI 10.17487/RFC9350, February 2023, <https://www.rfc-editor.org/info/rfc9350>.

[I-D.ietf-ippm-stamp-yang]
Mirsky, G., Min, X., Luo, W. S., and R. Gandhi, "Simple Two-way Active Measurement Protocol (STAMP) Data Model", Work in Progress, Internet-Draft, draft-ietf-ippm-stamp-yang-12, 5 November 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-ippm-stamp-yang-12>.

[IEEE.1588]
IEEE, "1588-2008 IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems", March 2008.

[IEEE802.1AX]
IEEE, "IEEE Standard for Local and Metropolitan Area Networks - Link Aggregation", IEEE Std 802.1AX-2020, DOI 10.1109/IEEESTD.2020.9105034, May 2020, <https://doi.org/10.1109/IEEESTD.2020.9105034>.

## Acknowledgments

The authors would like to thank Ianik Semco and Thierry Couture for their discussions on the use cases for Performance Measurement in Segment Routing.  The authors would also like to thank Greg Mirsky, Gyan Mishra, Xie Jingrong, Zafar Ali, Boris Hassanov, Ruediger Geib, Liyan Gong, Zhenqiang Li, Maria Matejka, William Hawkins, and Mike Koldychev for reviewing this document and providing useful comments and suggestions.  Additionally, Patrick Khordoc, Haowei Shi, Amila Tharaperiya Gamage, Pengyan Zhang, Ruby Lin, Senni Tan, and Radu Valceanu have helped improve the mechanisms described in this document.

## Contributors

The following people have substantially contributed to this document:

Daniel Voyer
Cisco Systems, Inc.
Email: davoyer@cisco.com

Navin Vaghamshi
Reliance
Email: Navin.Vaghamshi@ril.com

Moses Nagarajah
Telstra
Email: Moses.Nagarajah@team.telstra.com

Amit Dhamija
Arrcus
India
Email: amitd@arrcus.com

## Authors' Addresses

Rakesh Gandhi (editor)
Cisco Systems, Inc.
Canada
Email: rgandhi@cisco.com

Clarence Filsfils
Cisco Systems, Inc.
Email: cfilsfil@cisco.com

Bart Janssens
Colt
Email: Bart.Janssens@colt.net

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

Richard Foote
Nokia
Email: footer.foote@nokia.com