

Performance Delay and Liveness Monitoring in Segment Routing Networks

draft-gandhi-spring-sr-enhanced-plm-01

Rakesh Gandhi - Cisco Systems (rgandhi@cisco.com) - Presenter

Clarence Filsfils - Cisco Systems (cfilsfil@cisco.com)

Navin Vaghamshi - Reliance (Navin.Vaghamshi@ril.com)

Moses Nagarajah - Telstra (Moses.Nagarajah@team.telstra.com)

Agenda

- Requirements and Scope
- Summary
- Next Steps

Requirements and Scope

Requirements:

- Performance Delay & Liveness Monitoring in SR networks
 - ✓ End-to-end P2P/P2MP SR paths
 - ✓ Applicable to SR-MPLS/SRv6 data planes
 - ✓ Support ECMP SR paths
- Running single protocol for liveness detection and performance measurement in SR networks
 - ✓ Simplify deployment and reduce operational complexity
- No endpoint dependency
 - ✓ Stateless on endpoint (e.g. endpoint unaware of the probe protocol)
 - ✓ Higher scale and faster detection interval (e.g. packets not punted out of fast-path)

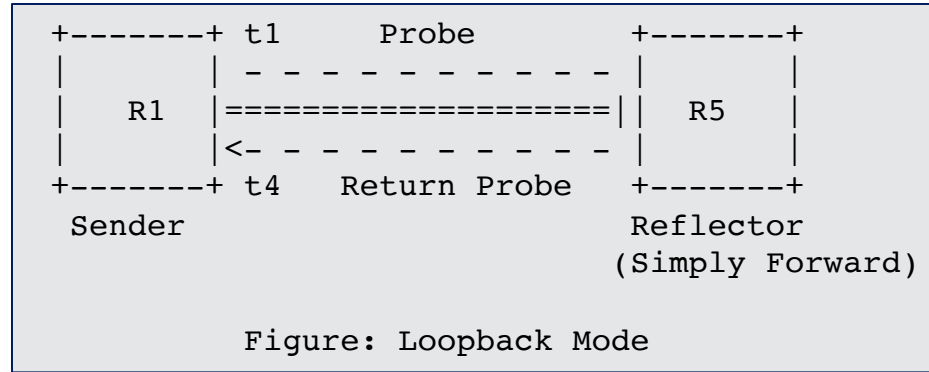
Scope:

- RFC 5357 (TWAMP Light) defined probe messages
- RFC 8762 (STAMP) defined probe messages

History of the Draft

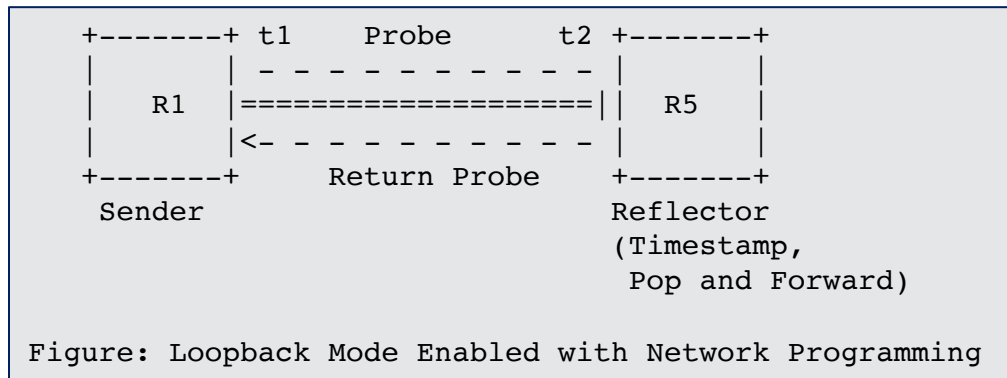
- March 2020
 - Draft was published
- April 2020
 - Presented *version 00* in IETF MPLS WG Virtual Meeting

Performance Delay and Liveness Monitoring of SR Policy



- Using PM probes (TWAMP Light/STAMP delay measurement messages) in Loopback Mode
- Probe messages sent using Segment List(s) of the SR Policy Candidate Path(s)
- Probe messages are not punted on the reflector out of fast-path
- Return path can be IP or SR
- Liveness failure is notified when consecutive N number of probe messages are not received back at the sender
- Round-trip delay ($t4 - t1$) metrics are notified when consecutive M number of probe messages have delay values exceed the configured thresholds

Enhanced Performance Delay and Liveness Monitoring of SR Policy



- Using PM probes in loopback mode enabled with network programming function
 - The network programming function optimizes the "operations of punt, add receive timestamp and inject the probe packet" on the reflector node
- Reflector node adds the receive timestamp in the payload of the received probe message without punting the message
 - Only adds the receive timestamp if the source address in the probe message matches the local node address
- Liveness failure is notified when consecutive N number of probe messages are not received back at the sender
- One-way delay ($t2 - t1$) metrics are notified when consecutive M number of probe messages have delay values exceed the configured thresholds

Probe Message with Timestamp and Forward Function

- Sender adds the Transmit Timestamp
- Reflector adds the Receive Timestamp at fixed offset locally provisioned (consistently in the network)
 - E.g. offset-byte 16 from the start of the payload

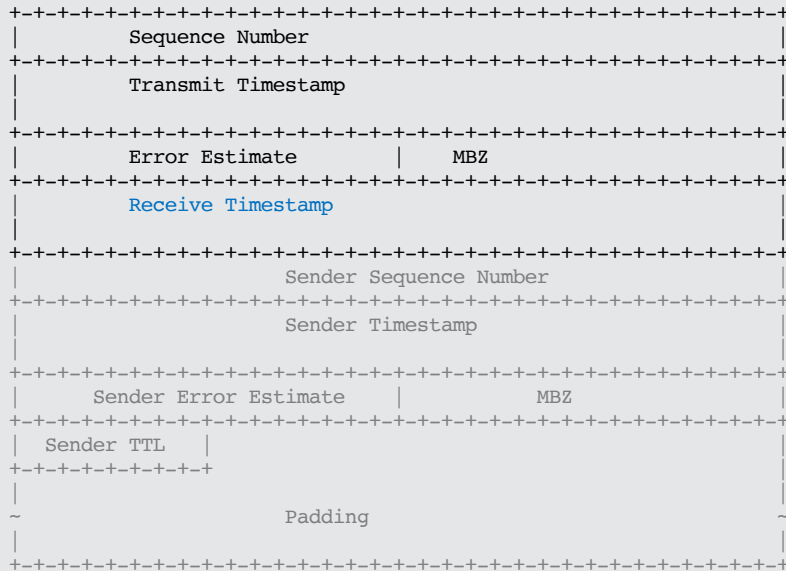


Figure: TWAMP Light Probe Message Format



Figure: STAMP Probe Message Format

SR-MPLS with Timestamp and Forward Function

```

0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                                     | TC | S |         TTL         |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
.
.
.
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                                     | TC | S |         TTL         |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                               Timestamp Label (TBA1)           | TC | S |         TTL         |
+-----+-----+-----+-----+-----+-----+-----+-----+
| IP Header                                                                |
.  Source IP Address = Reflector IPv4 or IPv6 Address                    .
.  Destination IP Address = Sender IPv4 or IPv6 Address                 .
.  Protocol = UDP                                                         .
.
+-----+-----+-----+-----+-----+-----+-----+-----+
| UDP Header                                                                |
.  Source Port = As chosen by Sender                                       .
.  Destination Port = User-configured Port                                .
.
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
| Payload as defined in Section 4.2.1 of RFC 5357                        |
| Payload as defined in Section 4.2 of RFC 8762                          |
.
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Example Probe Message with Timestamp Label for SR-MPLS

- Extended Special-purpose label (TBA1) is defined for Timestamp and Forward network function
- Reverse Path can be IP or SR-MPLS
- Source and Destination Addresses are swapped to represent the Reverse direction path

SRv6 with Timestamp and Forward Function

```

+-----+
| IP Header |
+-----+
| . Source IP Address = Sender IPv6 Address |
| . Destination IP Address = Next IPv6 Address |
| . |
+-----+
| SRH as specified in RFC 8754 |
| . <Segment List> |
| . END.TSF with Target SID |
| . |
+-----+
| IP Header |
+-----+
| . Source IP Address = Reflector IPv6 Address |
| . Destination IP Address = Sender IPv6 Address |
| . |
+-----+
| UDP Header |
+-----+
| . Source Port = As chosen by Sender |
| . Destination Port = User-configured Port |
| . |
+-----+
| Payload as defined in Section 4.2.1 of RFC 5357 |
| Payload as defined in Section 4.2 of RFC 8762 |
| . |
+-----+

```

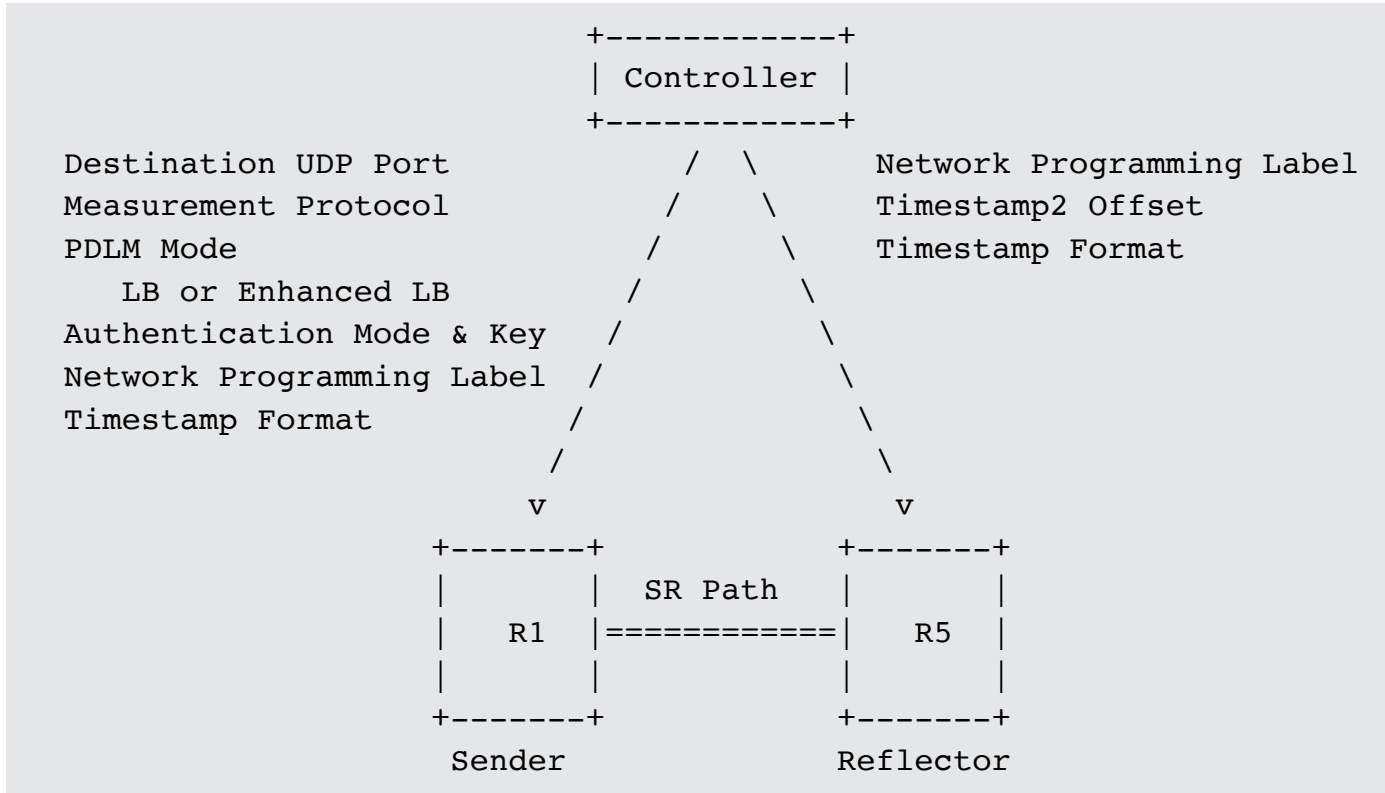
Example Probe Message with Endpoint Function for SRv6

- Endpoint Function END.TSF is defined for Timestamp and Forward and carried for the Reflector node SID
- Reverse path can be IP
 - Reflector node removes SRH
- Reverse path can be SR
 - Reverse direction SR path carried in SRH
 - Reflector node does not remove the SRH
- Source and Destination Addresses are swapped to represent the Reverse direction path in the inner IPv6 header

ECMP Support for SR Paths

- SR Paths can have ECMP between the ingress and transit nodes, between transit nodes and between transit and egress nodes.
- Sending PM probe queries that can take advantage of the hashing function in forwarding plane.
- Existing forwarding mechanisms are applicable to PM probe messages. Examples are:
 - For IPv4
 - Sweeping destination address in IPv4 header (e.g. 127/8) if return path is SR-MPLS
 - For IPv6
 - Sweeping flow label in IPv6 header

Example Provisioning Model



Next Steps

- Welcome your comments and suggestions
- Requesting WG adoption

Thank you