

Enhanced Performance Delay and Liveness Monitoring in Segment Routing Networks

draft-gandhi-spring-sr-enhanced-plm-03

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)

Richard Foote - Nokia (footer.foote@nokia.com)

Agenda

- Requirements and Scope
- Summary
- Next Steps

Requirements and Scope

Requirements:

- Performance Delay Monitoring & Liveness Monitoring in SR networks
 - ✓ End-to-end P2P/P2MP SR paths
 - ✓ Applicable to SR-MPLS/SRv6 data planes
- Running single protocol in SR networks
 - ✓ Simplify implementations and reduce development cost
 - ✓ Simplify deployment and reduce operational complexity
- No reflector dependency
 - ✓ Stateless on reflector (e.g. reflector unaware of the monitoring protocol)
 - ✓ State is in the probe message - spirit of SR
 - ✓ Higher scale and faster detection interval

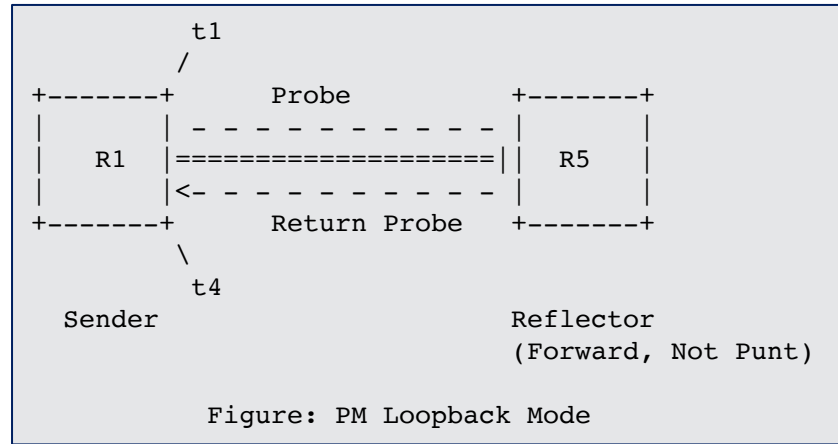
Scope:

- RFC 5357 (TWAMP Light) compatible probe message
- RFC 8762 (Simple TWAMP (STAMP)) compatible probe message

History of the Draft

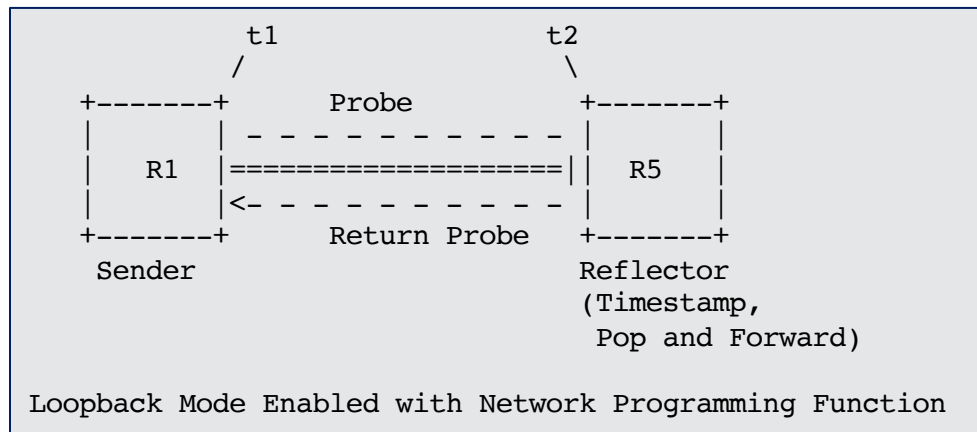
- March 2020
 - Draft was published
- April 2020
 - Presented version 00 in IETF 107 Virtual MPLS WG Meeting
- July 2020
 - Presented version 02 in IETF 108 Online SPRING WG meeting
- September 2020
 - Presented version 02 in MPLS WG Interim meeting

PM Probes in Loopback Mode for SR Policy



- Using PM delay measurement probe messages in Loopback Mode
- Probe messages are sent using Segment List(s) of the SR Policy Candidate Path(s)
- Probe messages are not punted on the reflector node out of fast-path in forwarding
- Reflector is agnostic to the monitoring protocol

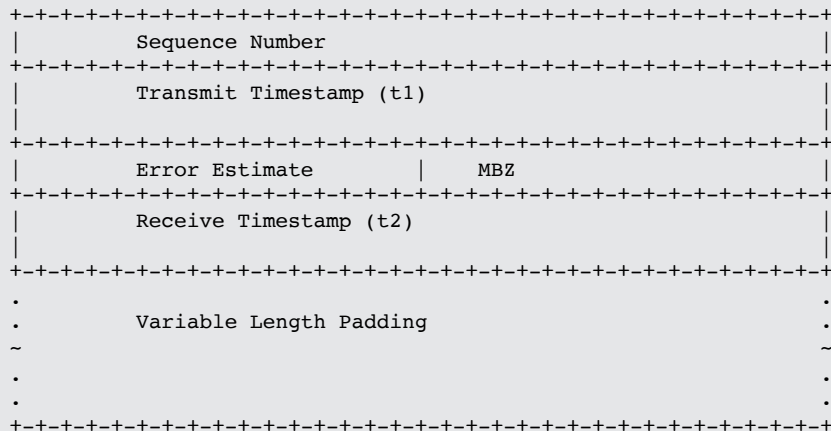
Enhanced Performance Delay and Liveness Monitoring of SR Policy



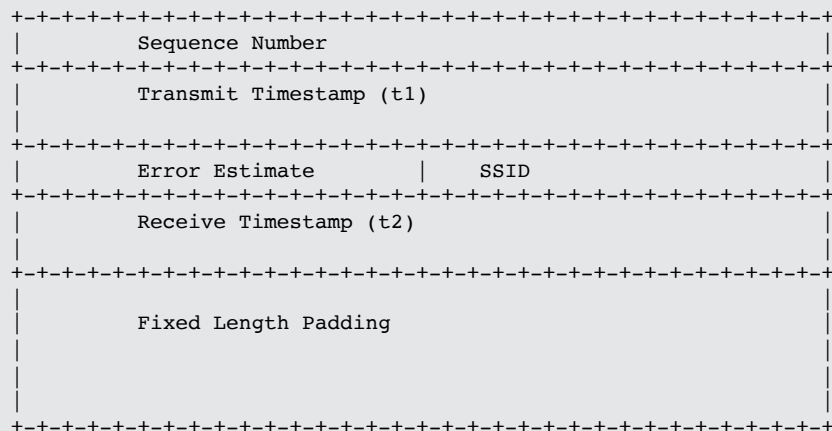
- Using PM probes in loopback mode enabled with network programming function
 - The network programming function optimizes the "operations of punt and inject the probe packet" on the reflector node
 - As probe packets are forwarded in fast-path, higher scale with faster interval is possible resulting in faster failure detection
- Reflector node adds the receive timestamp in the payload of the received probe message in the fast-path
 - Only adds the receive timestamp if the source address or destination address in the probe message matches the local node address
 - Ensure loopback probe packets return from the intended reflector node

Probe Messages for Network Programming Function

- Leverage existing TWAMP implementations and deployments using compatible probe message format
- Sender adds Transmit Timestamp (t1)
- Reflector adds Receive Timestamp (t2) at offset-byte location in payload
 - offset-byte 16 from the start of the payload, or
 - locally provisioned location (consistently in the network)



TWAMP Compatible Probe Message Format



STAMP Compatible Probe Message Format

SR-MPLS with Timestamp Label

```

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
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|          Label(1)          | TC | S |          TTL          |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
.
.
.
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|          Label(n)          | TC | S |          TTL          |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|          Extension Label (15)          | 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 .
.
+-----+
| UDP Header |
. Source Port = As chosen by Sender .
. Destination Port = As chosen by Sender .
.
+-----+
| Payload |
.
+-----+

```

Example Probe Message with Timestamp Label for SR-MPLS

- Timestamp label (TBA1) is defined for Timestamp, Pop and Forward function
- Reverse Path can be IP or SR-MPLS
- Source and Destination Addresses are swapped that represent the Reverse direction path

SRv6 with Timestamp Endpoint Function

```
+-----+
| IP Header                               |
. Source IP Address = Sender IPv6 Address .
. Destination IP Address = Destination IPv6 Address .
.                                           .
+-----+
| SRH as specified in RFC 8754           |
. <Segment List>                          .
. End.TSF (TBA2) with Reflector 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 = As chosen by Sender .
.                                           .
+-----+
| Payload                                |
.                                           .
+-----+
```

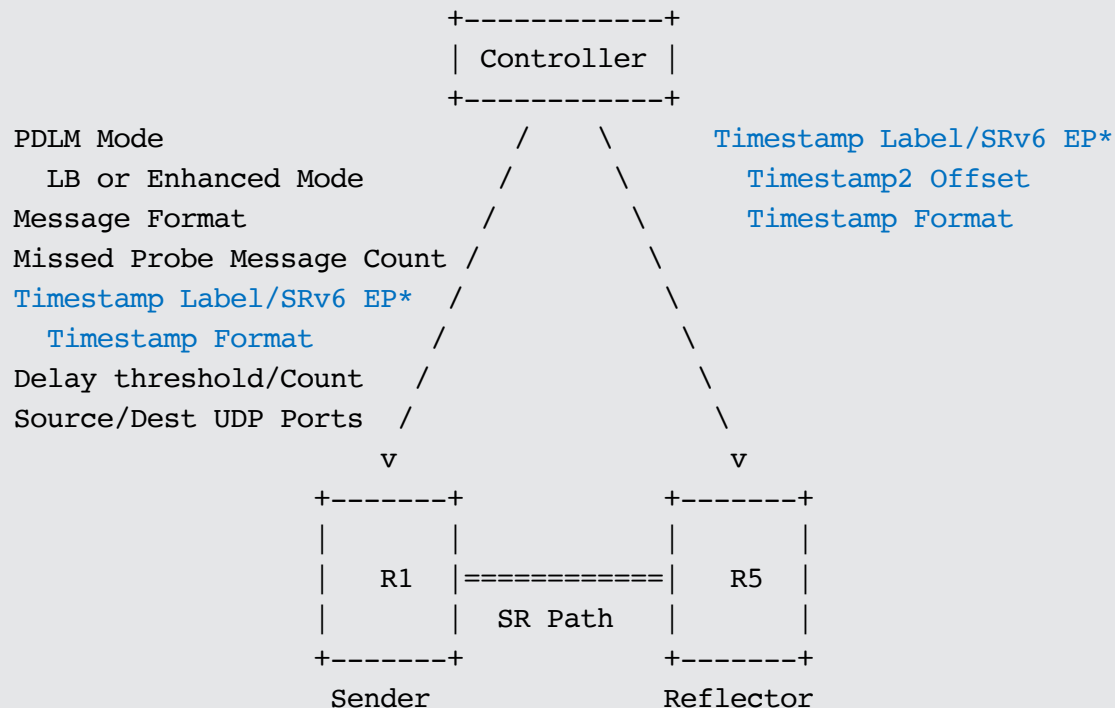
Example Probe Message with Timestamp Endpoint Function for SRv6

- Timestamp Endpoint Function End.TSF (TBA2) is defined for Timestamp and Forward and is carried with the Reflector node SID
- Reverse path can be IP
 - Reflector node removes SRH
- Reverse path can be SR
 - Reverse direction SR path Segment-list carried in SRH
 - Reflector node does not remove the SRH
- Source and Destination Addresses are swapped that represent the Reverse direction path in the inner IPv6 header

ECMP Support for SR Paths

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

Example Provisioning Model



* Provisioned,
Flooded/Signaled
or IANA Allocated

Example Provisioning Model

Notifications

- Delay metrics are notified when consecutive M number of probe messages have delay values exceed the configured thresholds
- Liveness failure (bring-down - loss of heart beats) is notified when consecutive N number of return probe messages are not received at the sender node
- Liveness success (bring-up - success of heart beats) is notified as soon as one or more return probe messages are received at the sender node

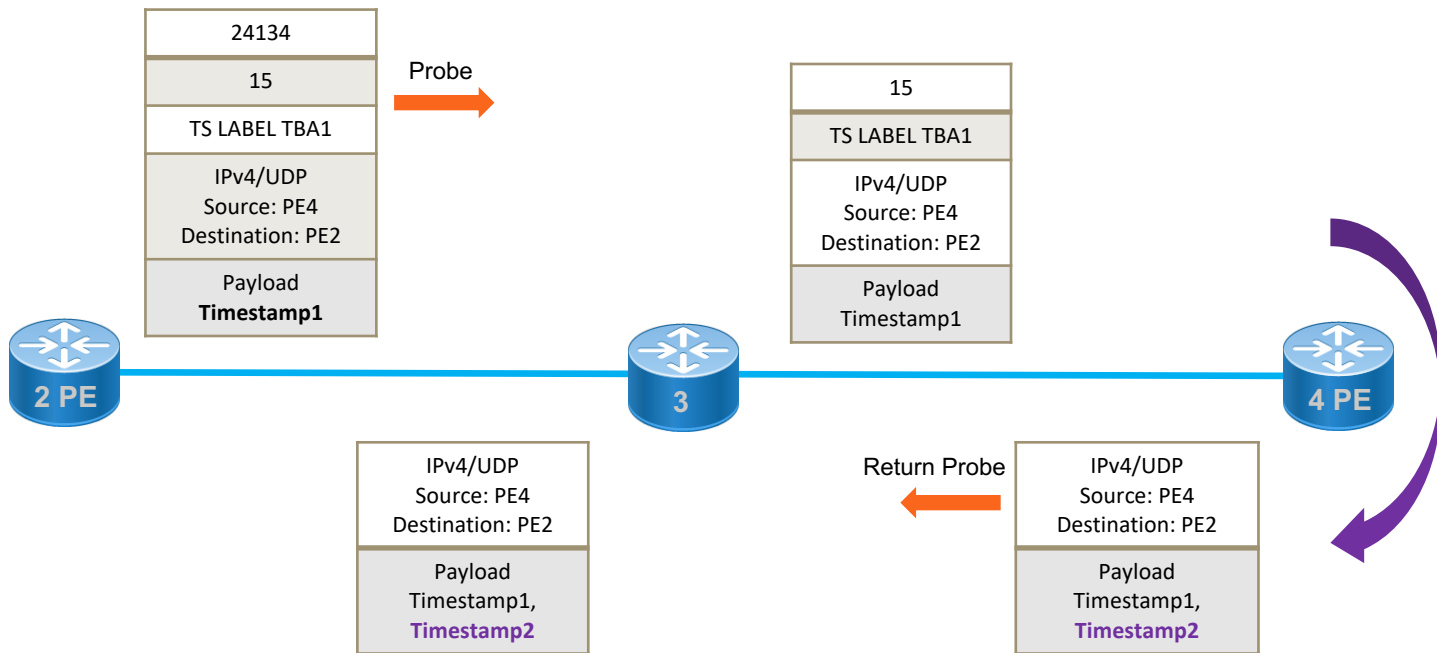
Next Steps

- Welcome your comments and suggestions
- Requesting SPRING WG adoption

Thank you

Backup

Loopback Mode with Timestamp and Forward for SR-MPLS Policy



Thank you