May 02, 2019

NETWORK PROGRAMMING FOR PERFORMANCE AND LIVENESS MONITORING IN SEGMENT ROUTING NETWORKS

Clarence Filsfils

Rakesh Gandhi

NETWORK PROGRAMMING FOR PERFORMANCE AND LIVENESS MONITORING IN SEGMENT ROUTING NETWORKS

AUTHORS:
Clarence Filsfils
Rakesh Gandhi

ABSTRACT

Techniques are described herein to define network programming functions for performance and liveness monitoring in Segment Routing (SR) and SRv6 networks. The network programming functions enable probe messages to run at significantly faster rates as punting probe messages to the control plane (slow path processing) and re-injecting them are not required. This enables hardware offloading for Performance Measurement (PM) sessions as well with liveness and PM probes combined. Network programming labels may be allocated from the global SR Global Block (SRGB) for SR Multiprotocol Label Switching (SR-MPLS) by a Software Defined Networking (SDN) controller. END functions are defined for SRv6 for performance delay, loss and liveness monitoring.

DETAILED DESCRIPTION

Segment-routing (SR) is a new technology that greatly simplifies network operations and makes networks SDN-friendly. SR is applicable to both Multiprotocol Label Switching (MPLS) (SR-MPLS) and Internet Protocol version 6 (SRv6) data planes. Built-in Performance Measurement (PM) is one of the essential requirements for the success of this technology. SR policies are used to steer traffic through a specific, user-defined path using a Segment ID (SID) list for Traffic Engineering (TE). In a SR network, there is a requirement to measure the end-to-end performance delay of customer traffic on SR policies to provide Service Level Agreements (SLAs).

For 5G networks, service providers are planning to use network slicing technology to deliver Ultra-Reliable Low-Latency Communication (URLLC) services for tele-medicine, online gaming, autonomous connected cars, and many other mission-critical applications. To provide these guaranteed services and achieve required SLAs, new sets of network functions must be enabled that can provide faster monitoring schemes to ensure there is no performance degradation due to congestion, faults, maintenance, or other issues.

In addition, network functions must detect performance degradation in the millisecond range.

PM probe messages may be used for both PM as well as for liveness monitoring. Liveness monitoring may involve one-to-one or one-plus-one path protection. Liveness monitoring enables verification of liveness of all end-to-end physical paths of an SR Policy to provide SLAs. The end-to-end liveness may be verified before activating the candidate path or the segment list(s) of the SR Policy in the forwarding table. The end-to-end liveness failure may be used to de-activate the active candidate path or the segment list(s) of the SR Policy in the forwarding table. The end-to-end liveness failure may be used to trigger path protection switchover to the standby candidate-path (one-to-one path protection) on the head-end node. The end-to-end liveness failure may also be used to trigger path protection switchover to the standby candidate-path (one-plus-one path protection) on the tail-end node for the Live-Live case.

There may be a Local Packet Transport Services (LPTS) Packet Per Second (PPS) limit for punting received packets. PM probe messages are punted in the control plane to process the query and response messages. The node may have a full mesh of SR Policies with destinations to different egress nodes in the network.

Figure 1 below illustrates an example reference topology. As shown, ingress node 2 may have SR Policies terminating on egress nodes 9, 3, 7, 6, 4, 8 and 5. Ingress node 2 may receive probe response messages from these egress nodes those are punted. In addition, ingress node 2 may receive probe query messages for the SR Policies originating from egress nodes 9, 3, 7, 6, 4 and 5 and terminating on ingress node 2, which are punted. The node may drop the received probe query and response messages if the incoming PPS rate exceeds the LPTS PPS limit of the platform for punting packets.

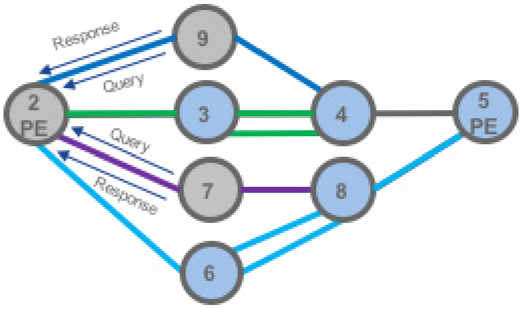


Figure 1

There may be a scale challenge associated with the SR Policy. PM probe messages are punted in the control plane slow path to process the query and response messages. For a multi-hop SR Policy, there may be Equal Cost Multi Pathing (ECMP) paths between ingress and transit nodes, between any two transit nodes, or between transit and egress nodes. As illustrated in Figure 2 below, this may result in a very large number of end-to-end atomic paths (e.g., 3x3x3=27 for three ECMP paths between two nodes) for the SR Policy. This "explosion" of end-to-end atomic paths can create a scale problem as a large number of PM sessions need to be created for delay measurement for the SR Policy.

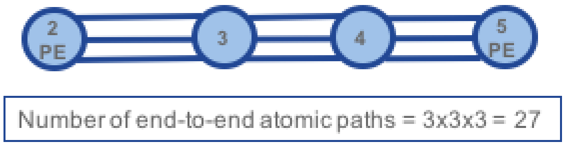


Figure 2

The solution described herein is defined using network programming that eliminates punting packets on the remote node with an enhanced loopback mode with a timestamp. This avoids sending two sets of probes, one for detecting liveness and one for measuring delay, while enabling scaling by a factor of two.

For performance delay monitoring, a network programming function "Timestamp, Pop and Forward" (TSF) may be defined that enables the hardware (micro-code / Application-Specific Integrated Circuit (ASIC)) to add the reception timestamp (T2) at a fixed offset location in the probe packet payload on the remote node. Similarly, for performance loss monitoring, the network function "Counter-stamp, Pop and Forward" (CSF) is defined, and for liveness monitoring, the network function "Address-stamp, Pop and Forward" (ASF) is defined. Further, the network programming function "Increment reception counter, Pop and Forward" for heart-bit monitoring enables providing Live-Live (one-plus-one path protection) on the tail-end node. The network functions "Queue-depth-stamp, Pop and Forward" and "5G network slice-signature-stamp, Pop and Forward" provide 5G network slice related monitoring functions.

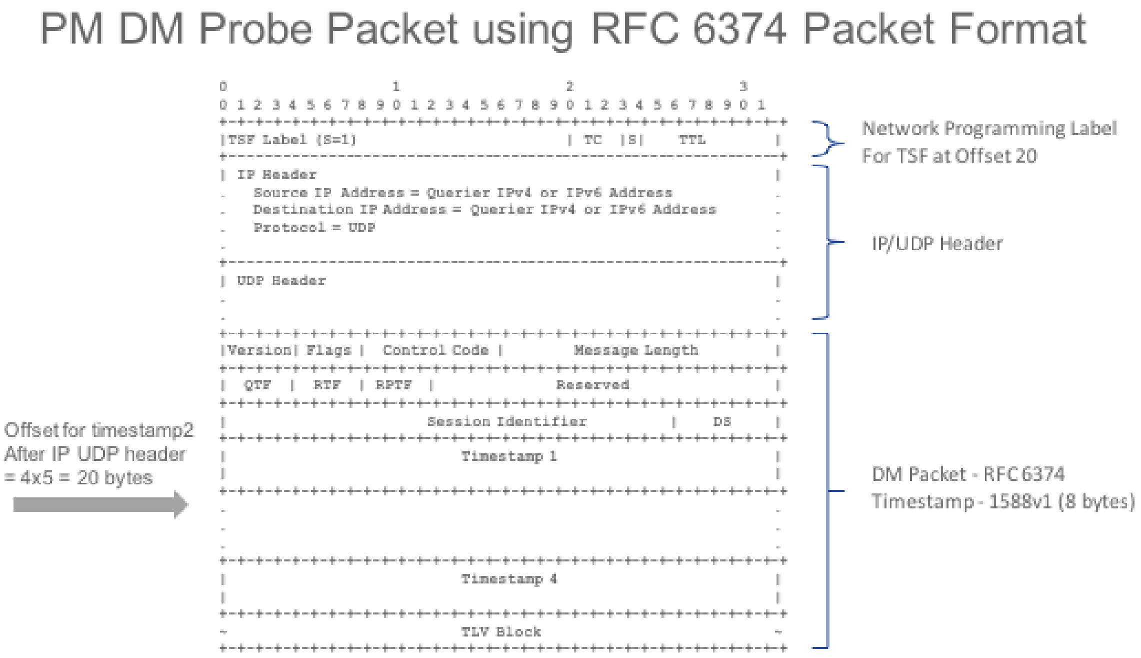
A PM-enabled (e.g., with a timestamp) adjacency / prefix SID may reduce the label stack size, particularly for segment-by-segment measurements with "Timestamp and Forward" behavior, for example. The probe messages are sent asynchronously in pipeline mode such that the querier does not wait for the response to return before sending the next probe query message.

The TSF network programming function may be implemented using a "Network Programming Label" for SR-MPLS and an "END Function" for SRv6. The TSF network programming function may enable the hardware to add a timestamp (e.g., at a fixed offset location K (bytes) from the End-of-Stack (EOS) Label, end of Internet Protocol (IP) / UDP Header, etc.), pop the TSF SID, and forward the packet. For end-to-end SR Policy delay measurement, the TSF network programming function is installed in hardware on the egress node.

The ingress node of the SR Policy sends the PM probe messages to the egress node with the TSF Label (SR-MPLS) or the END.TSF (SRv6) for the egress SID in the header and with destination address to itself, but via the egress node. The ingress node adds the transmission timestamp (T1) at a fixed location in the probe packet payload. The probe packet header contains the routing information in the header (e.g., MPLS header, IP header, SRv6 header, etc.) to return the probe packet back to the ingress node (in-band or out-of-band) and it contains both transmission and reception timestamps in the probe packet payload.

When the PM probe packet is received by the egress node, the hardware simply timestamps the probe packet, pops the SID, and forwards the packet using the MPLS header, SRv6 header, or IP header. The egress node adds the reception timestamp (T2) at a fixed offset K from the EOS Label (as an example) in the probe packet payload. The probe packets are not punted on the egress node control plane for slow-path processing. As such, the probe packets (replies) also do not need to be re-injected from the slow path. The PM probe querier may run in the control plane or in hardware (similar to Bidirectional Forwarding Detection (BFD) hardware offload) on the ingress node. The PM process may track the end-to-end delay for the SR Policy and trigger an action (such as protection switchover in hardware and/or re-optimization in the control plane) when the SLA is violated.

Figure 3 below illustrates a PM delay measurement probe packet using a packet format defined in Internet Engineering Task Force (IETF) Request For Comments (RFC) 6374.



PM DM Probe Packet using RFC 6374 Packet Format

```
 0                   1                   2                   3
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|TSF Label (S=1)                        | TC  |S|     TTL       |
+---------------------------------------------------------------+
| IP Header                                                     |
.   Source IP Address = Querier IPv4 or IPv6 Address            .
.   Destination IP Address = Querier IPv4 or IPv6 Address       .
.   Protocol = UDP                                              .
.                                                               .
+---------------------------------------------------------------+
| UDP Header                                                    |
.                                                               .
.                                                               .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version| Flags |  Control Code |         Message Length        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  QTF  |  RTF  | RPTF  |              Reserved                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Session Identifier          |     DS        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Timestamp 1                          |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
.                                                               .
.                                                               .
.                                                               .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Timestamp 4                          |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                          TLV Block                            ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Network Programming Label For TSF at Offset 20

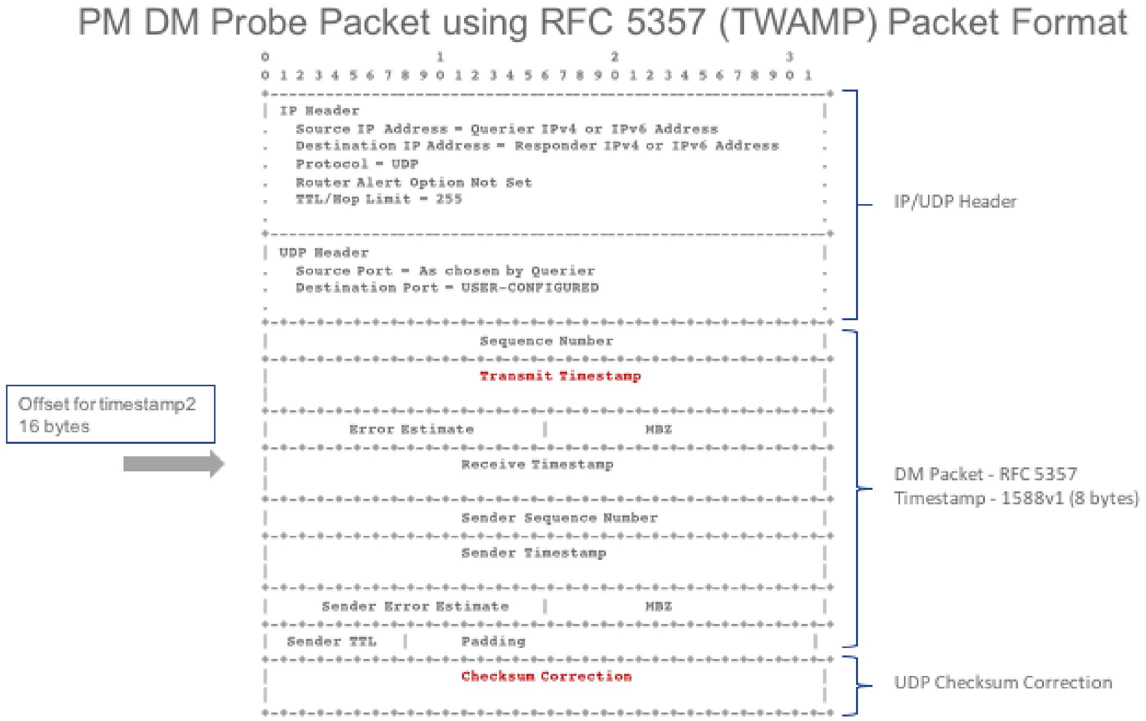
IP/UDP Header

Offset for timestamp2 After IP UDP header = 4x5 = 20 bytes

DM Packet - RFC 6374 Timestamp - 1588v1 (8 bytes)

Figure 3

Figure 4 below illustrates a PM delay measurement probe packet using a packet format defined in IETF RFC 5357.



PM DM Probe Packet using RFC 5357 (TWAMP) Packet Format

```
 0                   1                   2                   3
 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
+---------------------------------------------------------------+
| IP Header                                                     |
.   Source IP Address = Querier IPv4 or IPv6 Address            .
.   Destination IP Address = Responder IPv4 or IPv6 Address     .
.   Protocol = UDP                                              .
.   Router Alert Option Not Set                                 .
.   TTL/Hop Limit = 255                                         .
.                                                               .
+---------------------------------------------------------------+
| UDP Header                                                    |
.   Source Port = As chosen by Querier                          .
.   Destination Port = USER-CONFIGURED                          .
.                                                               .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Sequence Number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Transmit Timestamp                    |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Error Estimate        |              MBZ              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Receive Timestamp                       |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Sender Sequence Number                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Sender Timestamp                        |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Sender Error Estimate    |              MBZ              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Sender TTL  |      Padding                                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Checksum Correction                      |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

IP/UDP Header

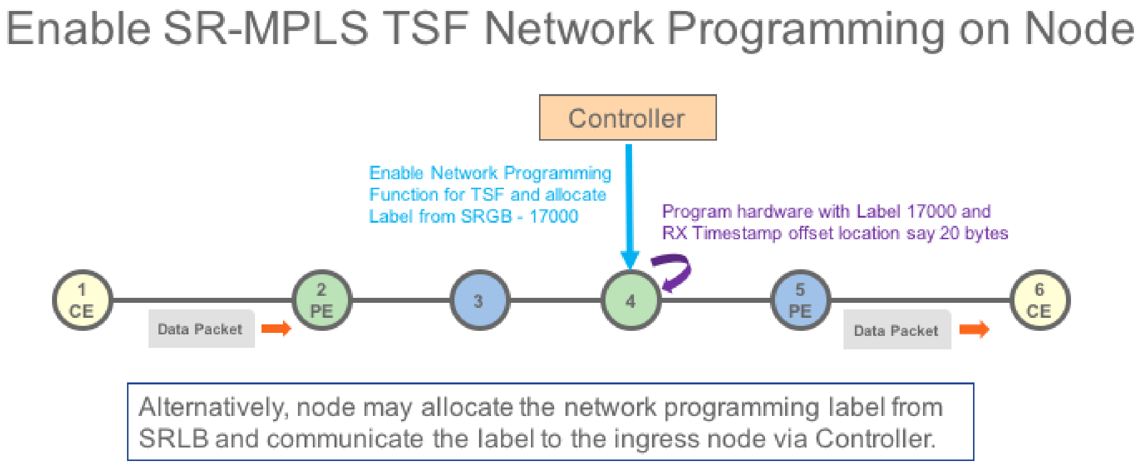
Offset for timestamp2 16 bytes

DM Packet - RFC 5357 Timestamp - 1588v1 (8 bytes)

UDP Checksum Correction

Figure 4

Figure 5 below illustrates an example process for enabling SR-MPLS TSF network programming on a node.



Enable SR-MPLS TSF Network Programming on Node

Controller

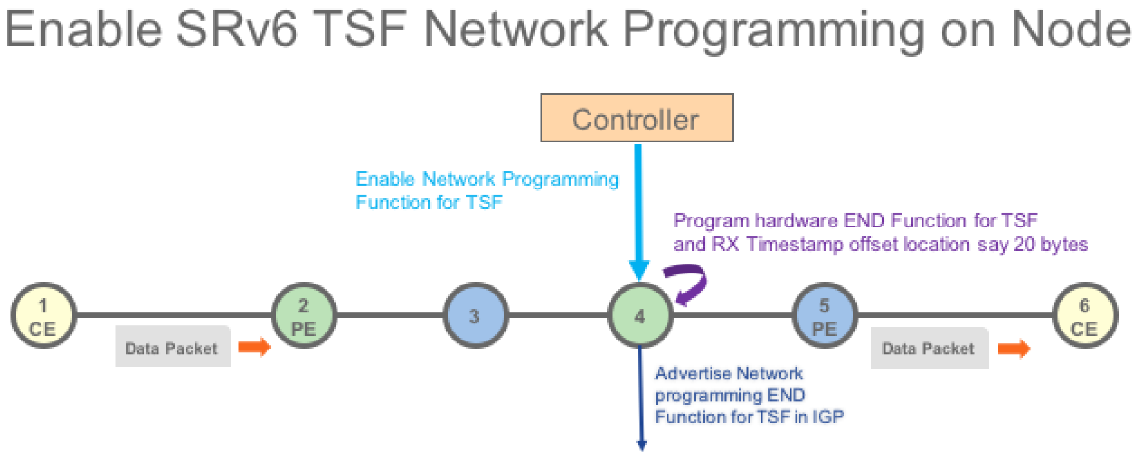
Enable Network Programming Function for TSF and allocate Label from SRGB - 17000

Program hardware with Label 17000 and RX Timestamp offset location say 20 bytes

Alternatively, node may allocate the network programming label from SRLB and communicate the label to the ingress node via Controller.

Figure 5

Figure 6 below illustrates an example process for enabling SRv6 TSF network programming on a node.



Enable SRv6 TSF Network Programming on Node

Controller

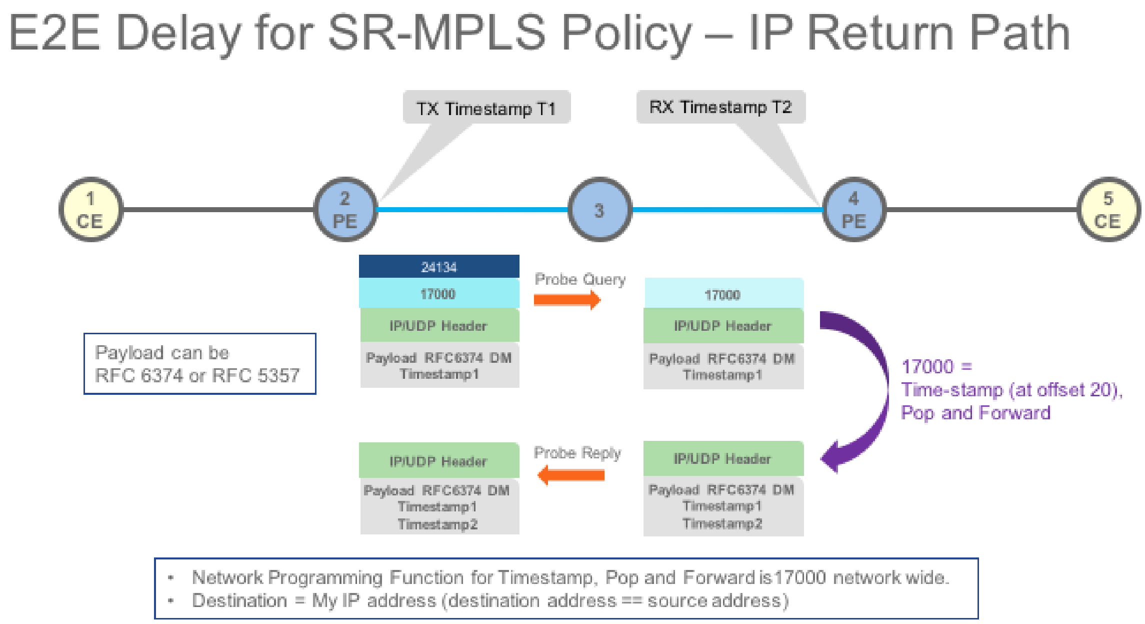
Enable Network Programming Function for TSF

Program hardware END Function for TSF and RX Timestamp offset location say 20 bytes

Advertise Network programming END Function for TSF in IGP

Figure 6

Figure 7 below illustrates an end-to-end delay for a SR-MPLS Policy for the IP return path.



E2E Delay for SR-MPLS Policy – IP Return Path

TX Timestamp T1
RX Timestamp T2
1 CE
2 PE
3
4 PE
5 CE
24134
17000
IP/UDP Header
Payload RFC6374 DM Timestamp1
Probe Query
17000
IP/UDP Header
Payload RFC6374 DM Timestamp1
IP/UDP Header
Payload RFC6374 DM Timestamp1 Timestamp2
Probe Reply
IP/UDP Header
Payload RFC6374 DM Timestamp1 Timestamp2

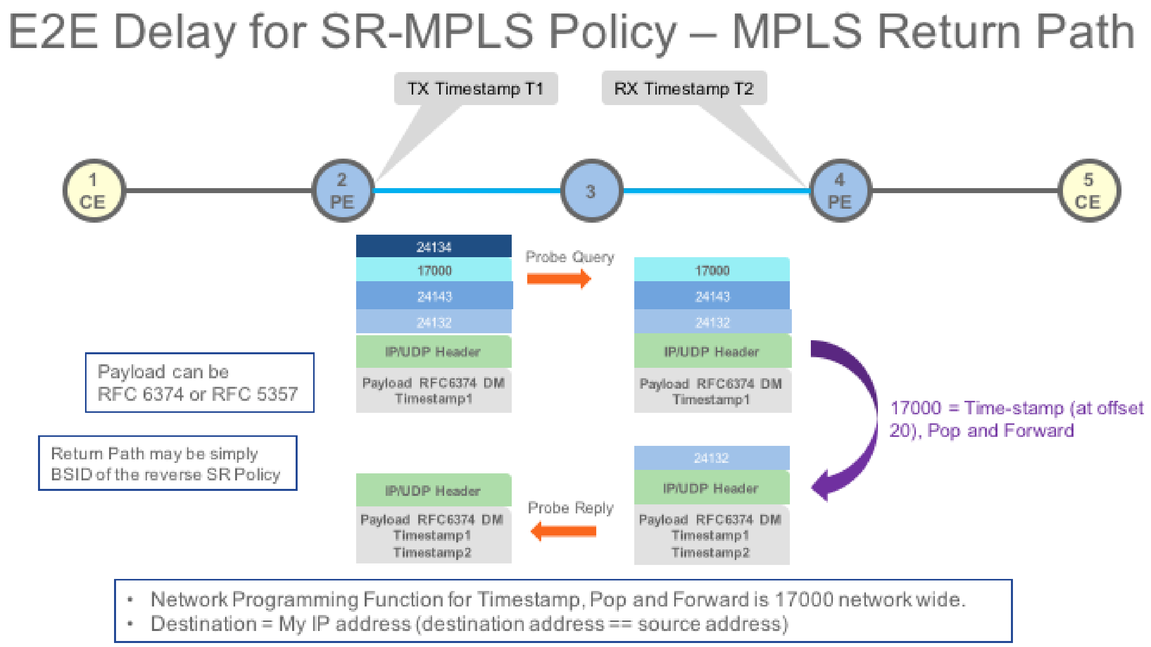
Payload can be RFC 6374 or RFC 5357

17000 = Time-stamp (at offset 20), Pop and Forward

- Network Programming Function for Timestamp, Pop and Forward is 17000 network wide.
- Destination = My IP address (destination address == source address)

Figure 7

Figure 8 below illustrates an end-to-end delay for a SR-MPLS Policy for the MPLS return path.



E2E Delay for SR-MPLS Policy – MPLS Return Path

Payload can be RFC 6374 or RFC 5357

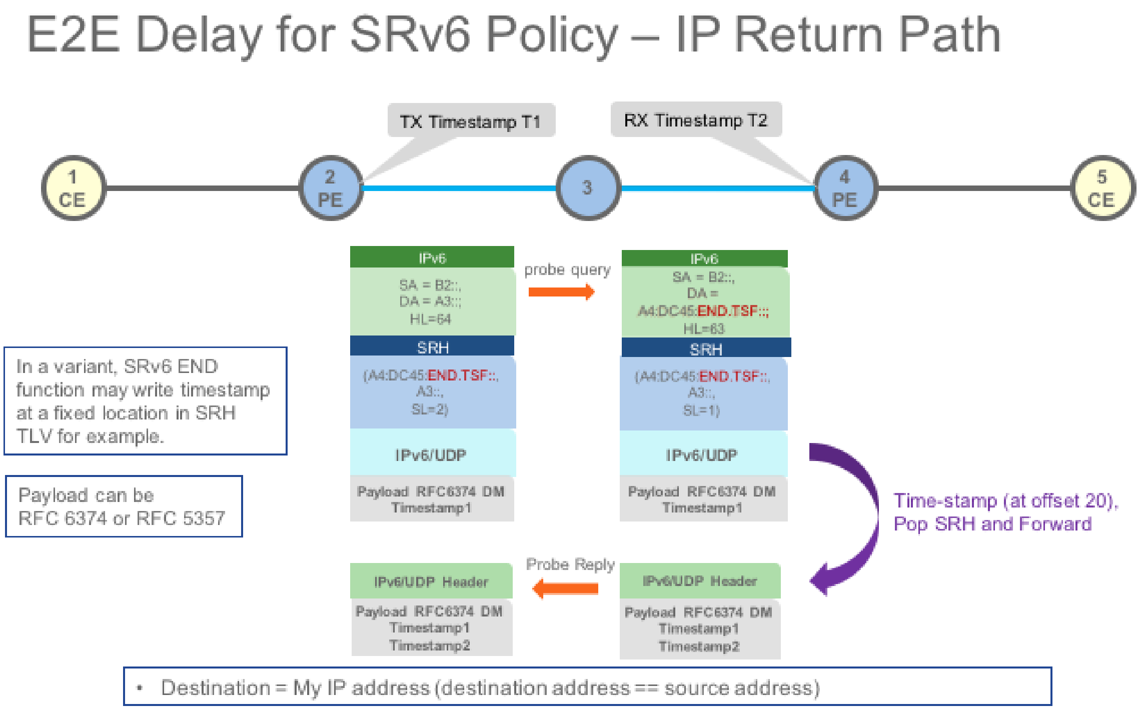
Return Path may be simply BSID of the reverse SR Policy

17000 = Time-stamp (at offset 20), Pop and Forward

- Network Programming Function for Timestamp, Pop and Forward is 17000 network wide.
- Destination = My IP address (destination address == source address)

Figure 8

Figure 9 below illustrates an end-to-end delay for a SRv6 Policy for the IP return path.



E2E Delay for SRv6 Policy – IP Return Path

In a variant, SRv6 END function may write timestamp at a fixed location in SRH TLV for example.

Payload can be RFC 6374 or RFC 5357

Time-stamp (at offset 20), Pop SRH and Forward

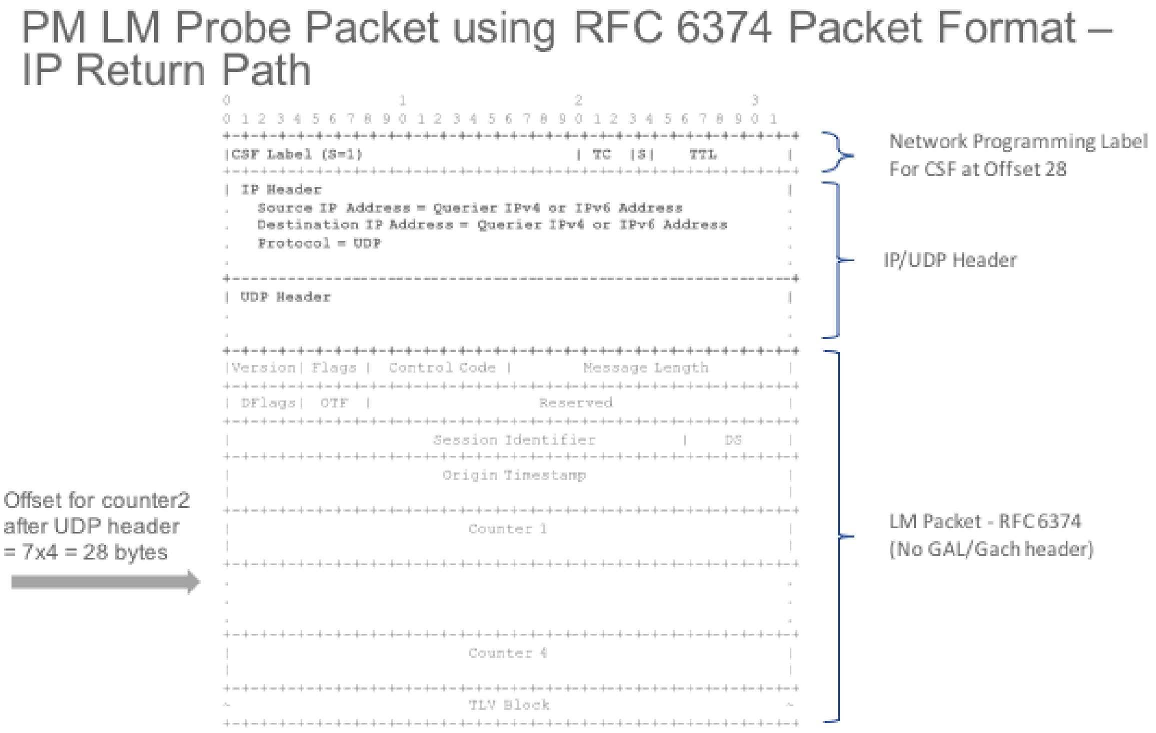
- Destination = My IP address (destination address == source address)

TX Timestamp T1
RX Timestamp T2
1 CE
2 PE
3
4 PE
5 CE
IPv6
SA = B2::, DA = A3::, HL=64
SRH
(A4:DC45:END.TSF::, A3::, SL=2)
IPv6/UDP
Payload RFC6374 DM Timestamp1
probe query
IPv6
SA = B2::, DA = A4:DC45:END.TSF::, HL=63
SRH
(A4:DC45:END.TSF::, A3::, SL=1)
IPv6/UDP
Payload RFC6374 DM Timestamp1
Probe Reply
IPv6/UDP Header
Payload RFC6374 DM Timestamp1 Timestamp2
IPv6/UDP Header
Payload RFC6374 DM Timestamp1 Timestamp2

Figure 9

A network programming function for performance loss monitoring is also described herein. Similar to performance delay monitoring, performance loss monitoring is implemented using the network programming function CSF. The CSF network programming function enables the hardware (in micro-code or ASIC) to counter-stamp (e.g., at an offset location K bytes from the EOS Label), pop the SID, and forward the packet. CSF may be based on the dual color method, where the reception counter value is based on the incoming SID counter on which the probe packet is received when using dual accounting SIDs. Alternatively, a different CSF may be used for each color.

Figure 10 below illustrates a PM loss monitoring probe packet using a packet format from IETF RFC 6374 for the IP return path.



PM LM Probe Packet using RFC 6374 Packet Format – IP Return Path

```
 0                   1                   2                   3
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|CSF Label (S=1)                        | TC  |S|     TTL       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| IP Header                                                     |
.   Source IP Address = Querier IPv4 or IPv6 Address            .
.   Destination IP Address = Querier IPv4 or IPv6 Address       .
.   Protocol = UDP                                              .
.                                                               .
+---------------------------------------------------------------+
| UDP Header                                                    |
.                                                               .
.                                                               .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version| Flags |  Control Code |         Message Length        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| DFlags|  OTF  |                  Reserved                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Session Identifier              |    DS     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Origin Timestamp                       |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Counter 1                           |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
.                                                               .
.                                                               .
.                                                               .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Counter 4                           |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                           TLV Block                           ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Network Programming Label For CSF at Offset 28

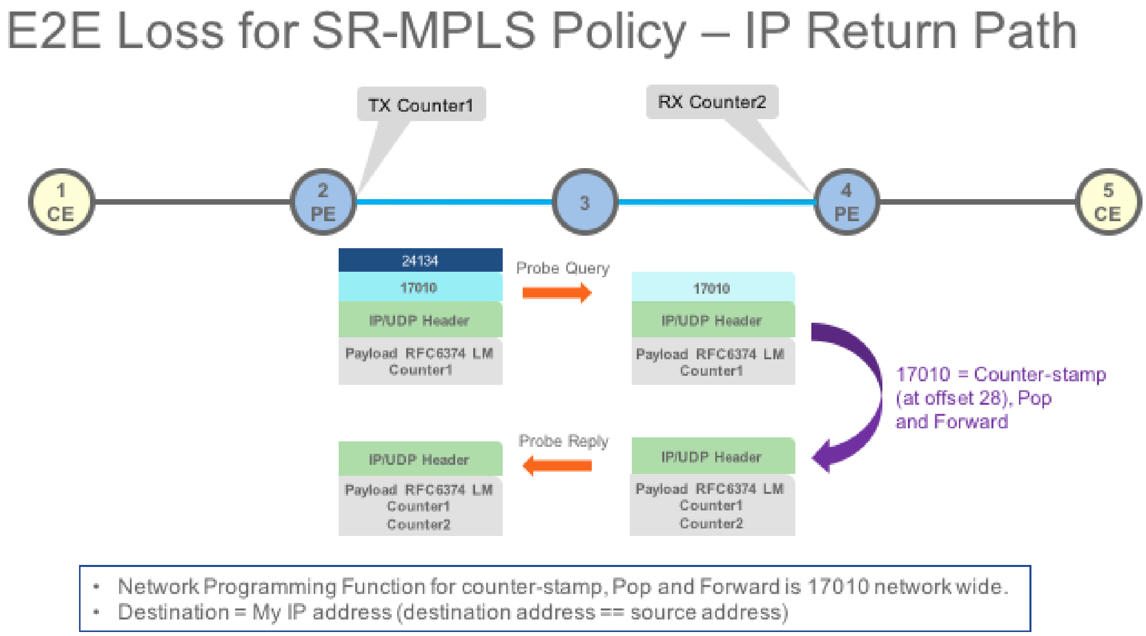
IP/UDP Header

LM Packet - RFC6374 (No GAL/Gach header)

Offset for counter2 after UDP header = 7x4 = 28 bytes

Figure 10

Figure 11 below illustrates an end-to-end delay for a SR-MPLS Policy for the IP return path.



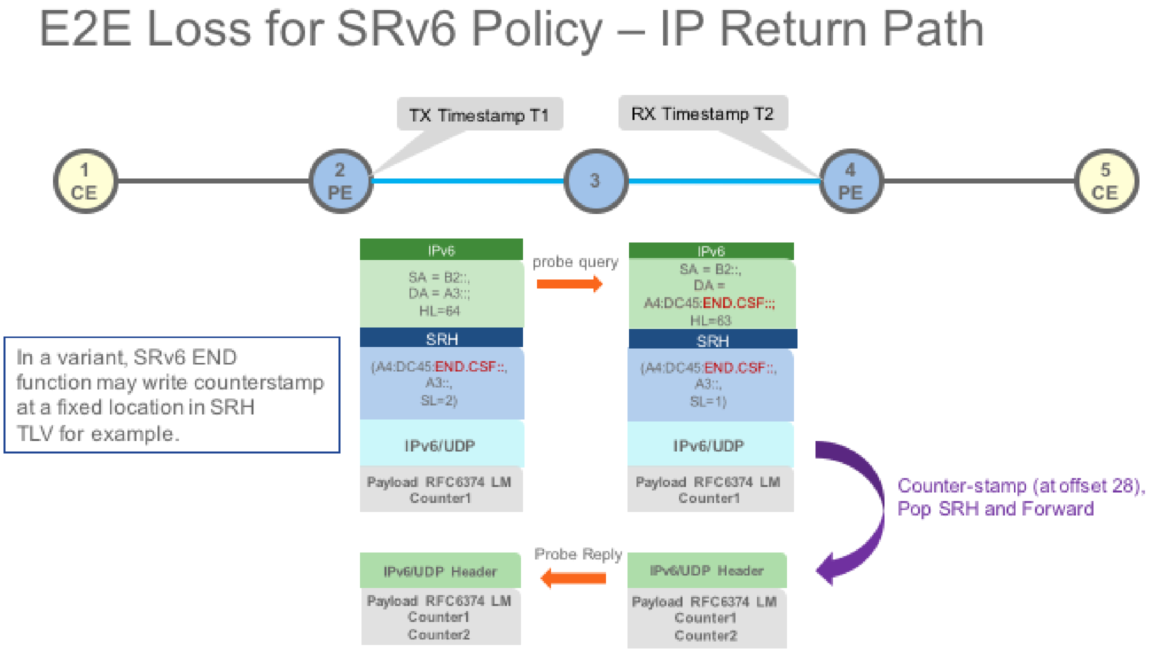
E2E Loss for SR-MPLS Policy – IP Return Path

17010 = Counter-stamp (at offset 28), Pop and Forward

- Network Programming Function for counter-stamp, Pop and Forward is 17010 network wide.
- Destination = My IP address (destination address == source address)

Figure 11

Figure 12 below illustrates an end-to-end loss for a SRv6 Policy for the IP return path.



E2E Loss for SRv6 Policy – IP Return Path

In a variant, SRv6 END function may write counterstamp at a fixed location in SRH TLV for example.

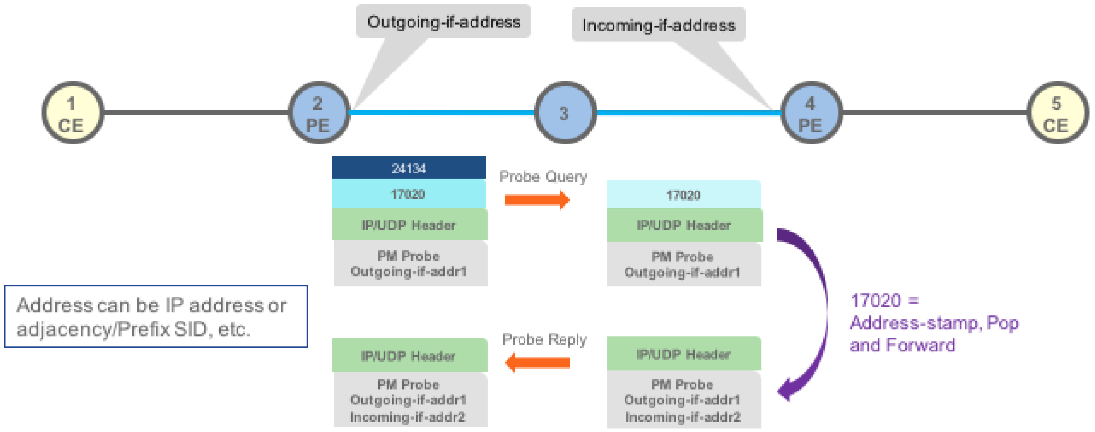
Counter-stamp (at offset 28), Pop SRH and Forward

Figure 12

A network programming function for performance liveness monitoring is also described herein. Similar to performance delay monitoring, the liveness monitoring is implemented using the ASF network programming function. The ASF network programming function enables the hardware (in micro-code or ASIC) to address-stamp (e.g., at an offset K bytes from the EOS Label), pop the SID, and forward the packet.

Figure 13 below illustrates end-to-end liveness monitoring for a SR-MPLS Policy for the IP return path.

E2E Liveness Monitoring for SR-MPLS Policy – IP Return Path


Outgoing-if-address
Incoming-if-address
1 CE
2 PE
3
4 PE
5 CE
24134
17020
IP/UDP Header
PM Probe Outgoing-if-addr1
Probe Query
17020
IP/UDP Header
PM Probe Outgoing-if-addr1
17020 = Address-stamp, Pop and Forward
Address can be IP address or adjacency/Prefix SID, etc.
IP/UDP Header
PM Probe Outgoing-if-addr1 Incoming-if-addr2
Probe Reply
IP/UDP Header
PM Probe Outgoing-if-addr1 Incoming-if-addr2

- Network Programming Function for address-stamp, pop and Forward is 17020 network wide.
- Destination = My IP address (destination address == source address)

Figure 13

A network programming function is also provided herein for segment-by-segment performance and liveness monitoring. For segment-by-segment performance monitoring, the network programming functions TSF, CSF, Increment Counter and Forward (INCF), Queue-depth-stamp, Pop and Forward (QSF), and ASF are enabled on all targeted nodes. Targeted nodes may add the reception TS, CS, QS, and/or AS at fixed offsets in the probe packet payload. Transit nodes may add the TS, CS and/or AS at different offsets in the payload using any suitable scheme, such as based on number of labels on the MPLS header, remaining SRv6 SIDs in the Segment Routing Header (SRH), etc.

In one example, the SR-MPLS network programming label may be allocated from the global SR Global Block (SRGB). A label value may be reserved in the SRGB or may be dynamically allocated (e.g., by an SDN controller). In one example, it may be an index in the SRGB. Separate label values may be used for different network programming PM functions (TSF, CSF, ASF, INCF, QSF, etc.). The network programming label is allocated domain-wide globally and not allocated per SR policy.

The SR-MPLS network programming label may be allocated by the node from the local SR Label Block (SRLB). In one example, it may be an index in the SRLB. In this case, the node floods the label or communicates the label to the ingress node via an SDN controller.

A node may advertise the adjacency or prefix SID with a timestamp enabled via Interior Gateway Protocol (IGP) (similar to a protected adjacency SID) with a flag defined for the timestamp. An ingress node may use the adjacency / prefix SID with the timestamp instead of a regular adjacency / prefix SID.

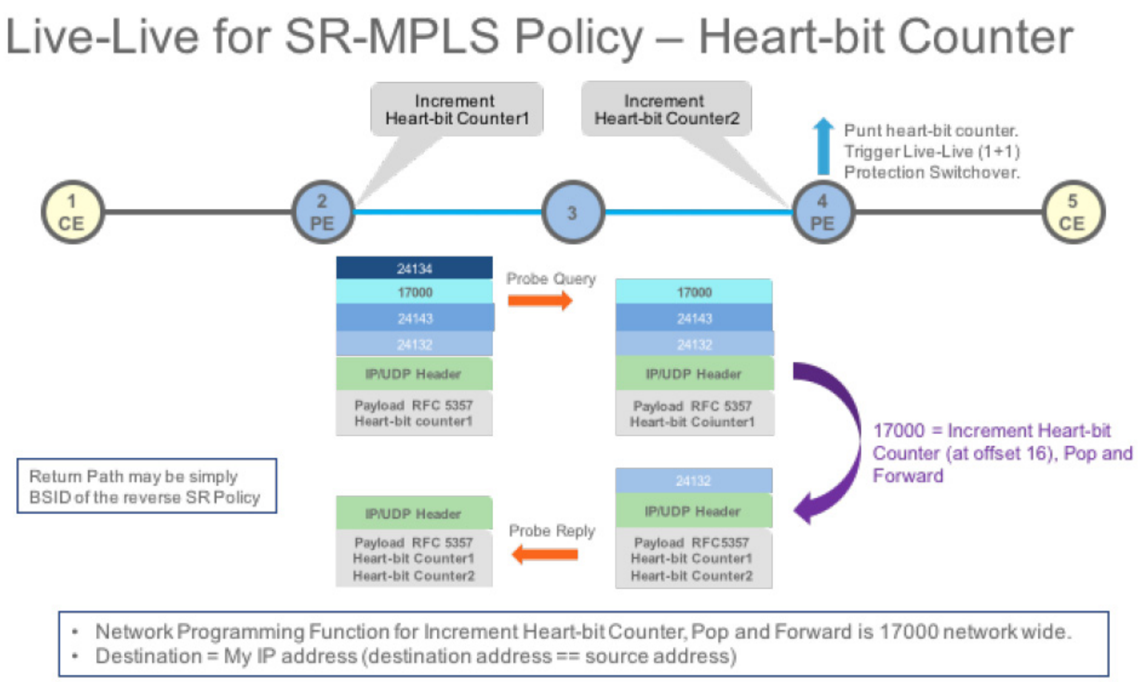
Further, the SID may provide the "Timestamp and Forward" function. When a node receives a packet with an adjacency / prefix SID with the timestamp, it timestamps the packet at a fixed known offset location in the packet and forwards the packet. Similarly, the node may advertise the adjacency / prefix SID for counter-stamping, address-stamping, or increment counter and forward behavior. This may reduce the size of the MPLS label stack by eliminating a separate network programming label, thereby providing significant advantages for segment-by-segment performance measurement.

SRv6 END functions for PM network programming functions (e.g., TSF, CSF, ASF, QSF, Slice-Stamp, Pop and Forward (SSF), INCF, etc.) may also be advertised via IGP by the node or programmed by a network controller. The SRv6 END functions may be used in the argument of the target SRv6 SID.

Heart-bit transmission and reception counters are defined on the PM probe querier and responder nodes. In addition to timestamping, counter-stamping, and/or address-stamping, the remote node also increments the reception heart-bit counter. Alternatively, a new network programming function (INCF) may be defined to increment the reception counter, pop the label, and forward the packet. When the transmission and reception heart-bit counters do not match, an alarm is raised about a potential fault on the SR policy path.

Heart-bit counters may be used by the tail-end node for live-live (one-plus-one) path protection. The head-end node sends traffic on both the active path and the standby candidate path for live-live path protection. The tail-end node starts a timeout timer for the heart-bit counter. If it is not incremented within the timeout, the tail-end node switches the traffic to the standby candidate path. The tail-end node hardware may punt the counter value at a periodic interval to the control plane, which is then used to trigger protection switchover on the tail-end node in case of missed heart-bits.

Figure 14 below illustrates an example heart-bit counter for live-live path protection for an SR-MPLS Policy.


Live-Live for SR-MPLS Policy – Heart-bit Counter

Increment Heart-bit Counter1

Increment Heart-bit Counter2

Punt heart-bit counter. Trigger Live-Live (1+1) Protection Switchover.

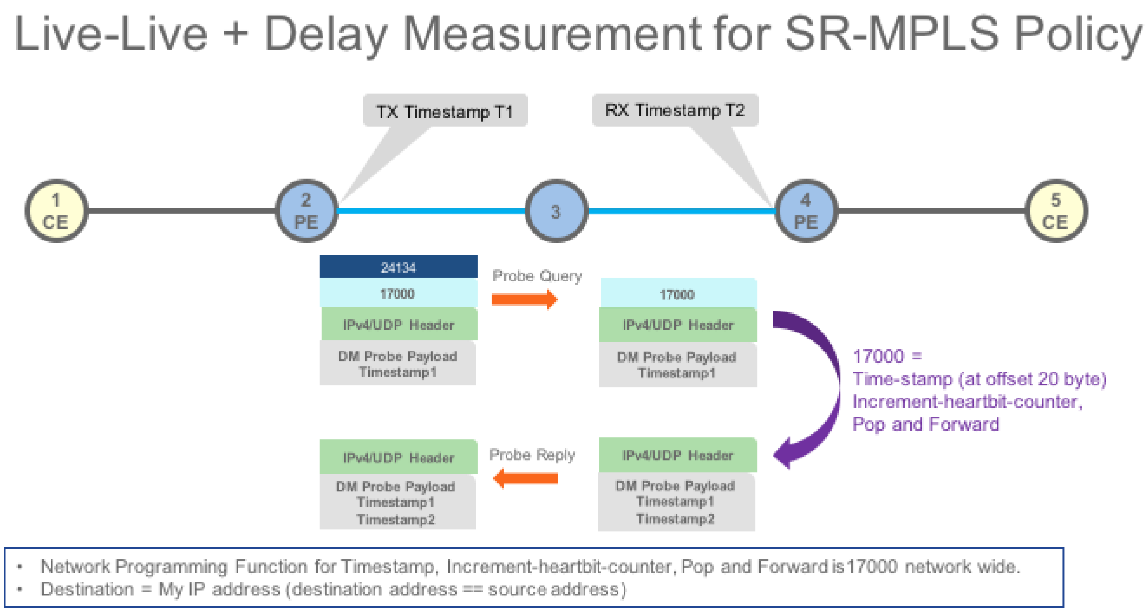
17000 = Increment Heart-bit Counter (at offset 16), Pop and Forward

Return Path may be simply BSID of the reverse SR Policy

- Network Programming Function for Increment Heart-bit Counter, Pop and Forward is 17000 network wide.
- Destination = My IP address (destination address == source address)

Figure 14

Figure 15 below illustrates the live-live and delay measurement for an example SR-MPLS Policy.


Live-Live + Delay Measurement for SR-MPLS Policy

TX Timestamp T1

RX Timestamp T2

17000 = Time-stamp (at offset 20 byte) Increment-heartbit-counter, Pop and Forward

- Network Programming Function for Timestamp, Increment-heartbit-counter, Pop and Forward is 17000 network wide.
- Destination = My IP address (destination address == source address)

Figure 15

In certain examples, SLA violation detection is also enabled. End-to-end delay is tracked using transmission timestamp t1 and reception timestamp t2 in the probe message.

When the delay values exceed the threshold, an alarm may be raised to trigger protection switchover in the control plane and data plane.

PM probe messages may be based on IETF RFC 6374 (MPLS-PM), IETF RFC 5357 (Two-Way Active Measurement Protocol (TWAMP)), etc. Payloads defined in these RFCs may be used to carry timestamps, counters, etc. This scheme may also be used with other messages such as Label Switched Path (LSP) Ping, Traceroute, etc. This scheme can also be used to reflect BFD packets.

Like performance delay monitoring, hardware queue monitoring may be implemented using the QSF network programming function. The QSF network programming function enables the hardware (e.g., in micro-code or ASIC) to queue-depth-stamp (e.g., at an offset location K bytes from the EOS label), pop the SID, and forward the packet. This field indicates the current length of the egress interface queue of the interface from which the packet is forwarded. It may also monitor the maximum value of the queue depth utilization for an interface.

As illustrated in Figure 16 below, like performance delay monitoring, the 5G network slice related information from hardware may be implemented using the SSF network programming function. The SSF network programming function enables the hardware (e.g., in micro-code or ASIC) to slice-depth-stamp (e.g., at an offset location K bytes from the EOS Label), pop the SID, and forward the packet. The 5G slice signature may be identification allocated hardware resource information (e.g., the particular queue being used) for the 5G slice, forwarding behavior for the packet, etc.

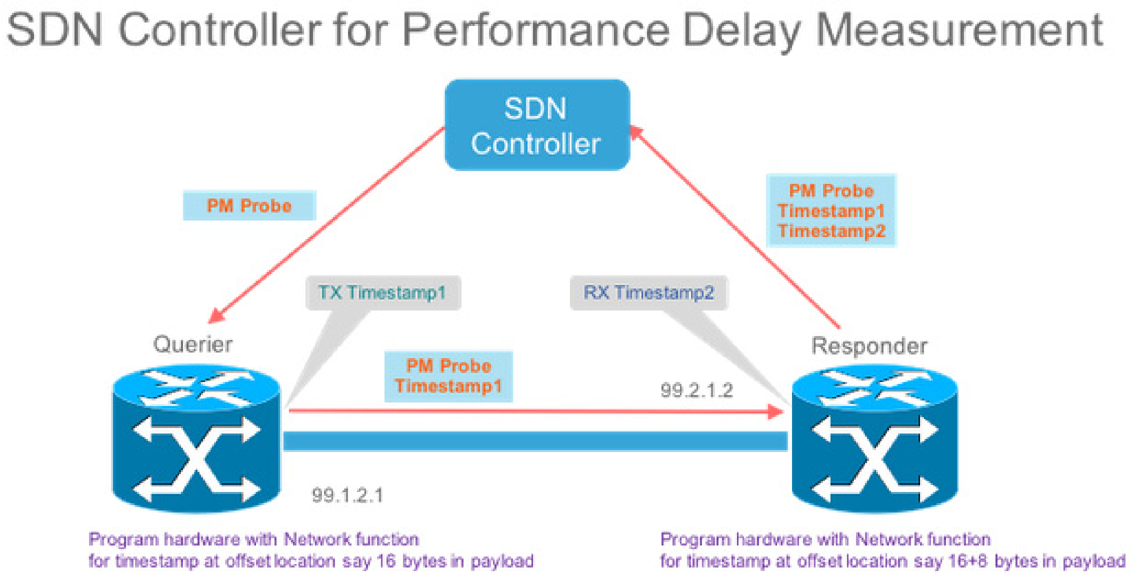

SDN Controller for Performance Delay Measurement
SDN Controller
PM Probe
PM Probe Timestamp1 Timestamp2
TX Timestamp1
RX Timestamp2
Querier
Responder
PM Probe Timestamp1
99.2.1.2
99.1.2.1
Program hardware with Network function for timestamp at offset location say 16 bytes in payload
Program hardware with Network function for timestamp at offset location say 16+8 bytes in payload

Figure 16

The techniques described herein enable running PMs and liveness monitoring using a single set of probes. Various network programming functions allow PM probes to run at faster rates because the network programming functions eliminate certain inefficiencies (e.g., punting the probe messages to the slow control plane path at the target node, re-injecting the reply back to the hardware, etc.). No complex control plane protocol support is required other than support for network programming functions on the egress node.

These techniques may be generically used to send probes and collect any data from any node in the network using the appropriate network programming function (e.g., collect queue size from node Z, write at offset location K in the payload, etc.). They may also be used to write information at other locations in the packet such as the SRH Type-Length-Value (TLV). Furthermore, the network programming functions are easy to implement in hardware microcode/ASIC.

Network programming enables multiple PM functions to be combined together (e.g., timestamp and counter-stamp). Using pipeline mode for sending probe queries allows for running probes at high rates and detecting faults faster than the round-trip-time. These solutions work for both the SR-MPLS and SRv6 data planes.

In summary, techniques are described herein to define network programming functions for performance and liveness monitoring in SR and SRv6 networks. The network programming functions enable probe messages to run at significantly faster rates as punting probe messages to the control plane (slow path processing) and re-injecting them are not required. This enables hardware offloading for PM sessions as well with liveness and performance measurement probes combined. Network programming labels may be allocated from the global SRGB for SR-MPLS by a SDN controller. END functions are defined for SRv6 for performance delay, loss and liveness monitoring.