Technical Disclosure Commons

Defensive Publications Series

March 22, 2019

ONE-WAY END-TO-END PATH DELAY MEASUREMENT FOR SR/SRV6 TRAFFIC ENGINEERING POLICIES WITHOUT CLOCK SYNCHRONIZATION IN SOFTWARE DEFINED NETWORKS

Amila Tharaperiya Gamage

Darren Dukes

Rakesh Gandhi

Radu Valceanu

Patrick Khordoc

ONE-WAY END-TO-END PATH DELAY MEASUREMENT FOR SR/SRV6 TRAFFIC ENGINEERING POLICIES WITHOUT CLOCK SYNCHRONIZATION IN SOFTWARE DEFINED NETWORKS

AUTHORS:
Amila Tharaperiya Gamage
Darren Dukes
Rakesh Gandhi
Radu Valceanu
Patrick Khordoc

ABSTRACT

Solutions are described herein that provide for one-way end-to-end path delay measurements for Segment Routing (SR) and SR with IPv6 (SRv6) Traffic Engineering (TE) policies without clock synchronization in Software Defined Networks (SDNs). A centralized solution is provided using a controller while a distributed solution is provided without using a controller.

DETAILED DESCRIPTION

The ability to measure one-way end-to-end delay for a SR Policy is a necessity for Service Level Agreement (SLA) assurance by network operators. Previous solutions have all required clock synchronization among nodes along a path, which is known to be complex to administer and maintain, or have measured bidirectional delay over symmetric forward and reverse paths. Defined herein are methods and an apparatus to measure one-way end-to-end delay for an SR Policy without clock synchronization between nodes.

SR Policies are expected to satisfy certain end-to-end delay constraints as defined in SLAs. For example, services such as tele-medicine, online gaming, stock market trading, and many mission critical applications have strict end-to-end delay bounds. Further, SR technology is planned to be used with network slicing to provide end-to-end low latency services in 5G networks.

End-to-end delay experienced by traffic varies with time due to variations of traffic loads (e.g., queue lengths) at routers on an end-to-end path. When end-to-end delay experienced by traffic violates the delay constraints of SLAs, such violations are expected to be detected and corrected using another path within a sub-second interval.

Satisfaction of end-to-end delay constraints is achieved via path calculation and verification. For path calculation, end-to-end paths of SR policies are calculated based on link delay metrics. Individual link delays are measured as a part of Performance Measurement (PM) feature. Since path calculation does not consider routers' inbound delay, which varies with traffic loads, in the verification phase the one-way end-to-end delay is periodically measured to ensure satisfaction of the delay constraints. The method proposed herein is used in the verification phase.

To measure one-way end-to-end path delay, synchronization of clocks at policy head-end and at tail-end routers is required. This can be achieved using Precision Time Protocol (PTP), as defined in the Institute of Electrical and Electronics Engineers (IEEE) 1588 protocol, or based on Global Positioning System (GPS) signaling or Network Time Protocol (NTP) in hardware. However, many routing platforms do not support PTP and network operators are reluctant to enable PTP due to additional operational overhead. Additionally, GPS-based clock synchronization mechanisms cannot be used due to weak GPS signal reception at indoor locations where routers are usually installed. For these reasons, clock synchronization becomes an issue which prevents reliable one-way end-to-end path delay measurement.

Typically for a link delay measurement, a one-way delay can be computed as a two-way delay divided by two as one can typically assume symmetrical delay in both directions. However, this assumption is not valid for SR Policies and one-way delay cannot be computed as a two-way delay divided by two since the traffic pattern in the reverse direction (and on the transit nodes) may be very different than the forward direction.

Two solutions for solving this problem are described herein: a centralized solution and a distributed solution. For the centralized solution, each node calculates clock offsets of its adjacent nodes. The calculated clock offsets are then sent to a Software Defined Network (SDN) controller, such as Path Computation Element (PCE), for calculating clock offsets of tail-end nodes. The SDN controller learns clock offsets among adjacent routers and then calculates end-to-end clock offset for calibrating a one-way end-to-end delay measurement. Use of an end-to-end clock offset to calibrate measured one-way end-to-end delay eliminates the requirement of synchronizing clocks. Calculation of the end-to-end clock offset by adding clock offsets of the routers on an end-to-end path eliminates the requirement of symmetric forward and reverse paths. Thus, this approach is simple, novel, and disproportionately impactful on the operational cost of measuring delay in a network.

For the distributed solution, a head-end node of an SR policy sends a clock offset query message along the SR Policy's end-to-end path. The query message is processed at each transit node. Once a transit node receives this message, it calculates a clock offset between itself and its upstream node using delay measurement probe packets. Each transit node then sends the calculated clock offsets to the head-end. After the head-end node receives all clock offsets, the tail-end's clock offset with respect to the head-end is calculated.

Solutions described herein only measure clock offset of adjacent nodes. When a link between two adjacent nodes is considered, the link delay only consists of the signal propagation delay in the link (e.g., propagation delay in fiber). Thus, link delays in forward and reverse directions are the same (i.e., symmetric link delays). Therefore, accurate clock offsets among adjacent nodes can be calculated using probe packets. Due to this reason, the proposed methods calculate clock offsets of adjacent nodes, and then add the calculated clock offsets along the end-to-end path to obtain clock offset of tail-end node with respect to head-end node.

To measure clock offset and link delay, probe packets are used and are timestamped at transmitting and receiving nodes. The transmitting node timestamps probe packets at the egress interface just before transmitting, and the receiving node timestamps probe packets at the ingress interface as soon as they are received.

For example, to calculate clock offset between two adjacent nodes, four timestamps are collected by sending probe packets over the link. The same four timestamps are used for calculating the link delay as well. These timestamps correspond to: 1) timestamp at egress interface of probe sending node, 2) timestamp at ingress interface of probe receiving node, 3) timestamp at egress interface of reply-probe sending node, and 4) timestamp at ingress interface of reply-probe receiving node. It should be noted that calculating clock offset between two adjacent nodes from both directions may not offer an improvement since all four timestamps are used for calculating clock offsets and that link delay of a link between two adjacent nodes is symmetric.

In the final step of the methods described herein, one-way end-to-end delay from head-end to tail-end is measured, and calibrated using the tail-end's clock offset with respect to the head-end. This step cannot be performed from both directions as end-to-end path delays on forward and reverse directions are not the same.

Centralized Solution

An overview of the method for determining one-way end-to-end delay measurement using the centralized solution is now provided. As noted above, the centralized solution involves an SDN controller. For the centralized solution, a one-way end-to-end path delay measurement that does not require synchronizing router clocks can be achieved as follows:

- Calculating relative clock offsets for every pair of connected routers in a domain
  - This can be performed utilizing the method defined in Request For Comments (RFC) 6374
- Sending the relative clock offsets to an SDN controller
- For an SR Policy (P) with a headend (H) and an endpoint (E), the SDN controller computes the clock offset between H and E (which is referred to herein using the term "COhe") as follows:
  - COhe = sum of clock offsets of nodes along any path between H and E
- For an SR Policy (P) with a headend (H) and endpoint (E), measure the one-way uncalibrated delay for a packet sent via P (which is referred to herein using the term "UDhe") as defined in RFC 6374
  - UDhe is not accurate as the clocks of H and E are not synchronized
  - Calculate the one-way calibrated delay for packets sent via P (which is referred to herein using the term "Dhe") as:
    - Dhe = UDhe – COhe

Novelty of the centralized solution lies in its simplicity. In this solution, the SDN controller learns clock offsets among adjacent routers, and then calculates end-to-end clock offset for calibrating one-way end-to-end delay measurement. Use of end-to-end clock offset to calibrate measured one-way end-to-end delay eliminates the requirement of synchronizing clocks. Calculation of end-to-end clock offset by adding clock offsets of the routers on end-to-end path eliminates the requirement of symmetric forward and reverse paths. Thus, this approach is simple, novel, and disproportionately impactful on the operational cost of measuring delay in a network.

Consider an example for the centralized solution in which details of the proposed one-way end-to-end path delay measurement method can be further explained.

Step 1: For the first step, each router determines relative clock offsets of its neighboring routers using the timestamps collected during link delay measurements. This relies on a simple fact that the delay of a link between any two nodes is symmetrical. Consider an example as shown in Figure 1, below, in which features associated individual link delay measurements are illustrated.

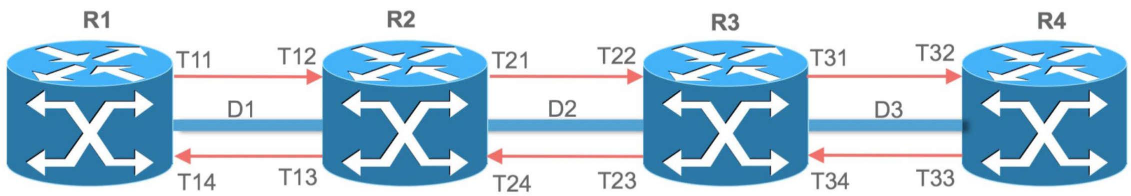

Figure 1

For Figure 1, an SR policy with an end-to-end path from R1 to R4 is configured at router R1. Figure 1 also shows timestamps taken during the link delay measurements. In general, when a router Rx measures delay of the link between Rx and Ry, it collects Tx1, Tx2, Tx3 and Tx4.

For example, when R1 measures delay of the link between R1 and R2, it collects T11, T12, T13 and T14 as described in RFC 6374. When R1 measures the delay of the link between R1 and R2, it collects the four timestamps as follows. First, R1 sends a probe packet to R2. This packet is timestamped at egress of R1 and at ingress of R2. Then, R2 sends that packet back to R1. When the packet is returned, it is again timestamped at egress of R2 and at ingress of R1. In the example involving R1, the returned probe packet has four timestamps as follows:

T11: R1's timestamp at the egress

T12: R2's timestamp at the ingress

T13: R2's timestamp at the egress

T14: R1's timestamp at the ingress

Continuing with Step 1, each router then calculates relative clock offsets of its neighboring routers. For this calculation, let COij denote the relative clock offset of router Ri with respect to router Rj. Accordingly, each router can calculate relative clock offsets of its neighboring routers as follows:

Based on the measurements, T12 and T14 are calculated as:

T12 = T11 + D1 + CO21

T14 = T13 + D1 - CO21

where D1 is the actual (i.e., propagation) delay of the link between R1 and R2 and, similarly, D2 and D3 are the delays of the links between R2 and R3, and R3 and R4, respectively.

From those two equations, CO21 can be calculated as:

CO21 = (T12 - T11)/2 - (T14 - T13)/2

Similarly, CO32 and CO43 can be calculated as:

CO32 = (T22 - T21)/2 - (T24 - T23)/2

CO43 = (T32 - T31)/2 - (T34 - T33)/2

<u>Step 2</u>: For the second step, the routers send the relative clock offset information to an SDN controller, as shown in Figure 2, below. In this step, Relative clock offset (CO) information is sent to the SDN controller, along with the link delay measurements, via telemetry.

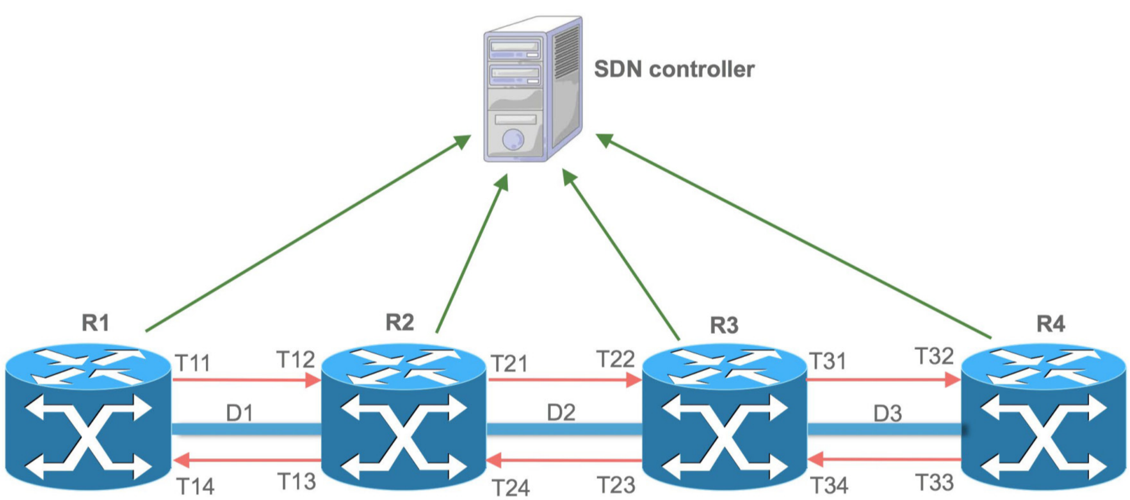

Figure 2

<u>Step 3</u>: For the third step, an accurate one-way delay of an end-to-end path of a SR policy is determined by combining measured end-to-end delay and the relative clock offset of the policy tail-end router with respect to the policy head-end router. Details of the third step are illustrated in Figure 3, below.

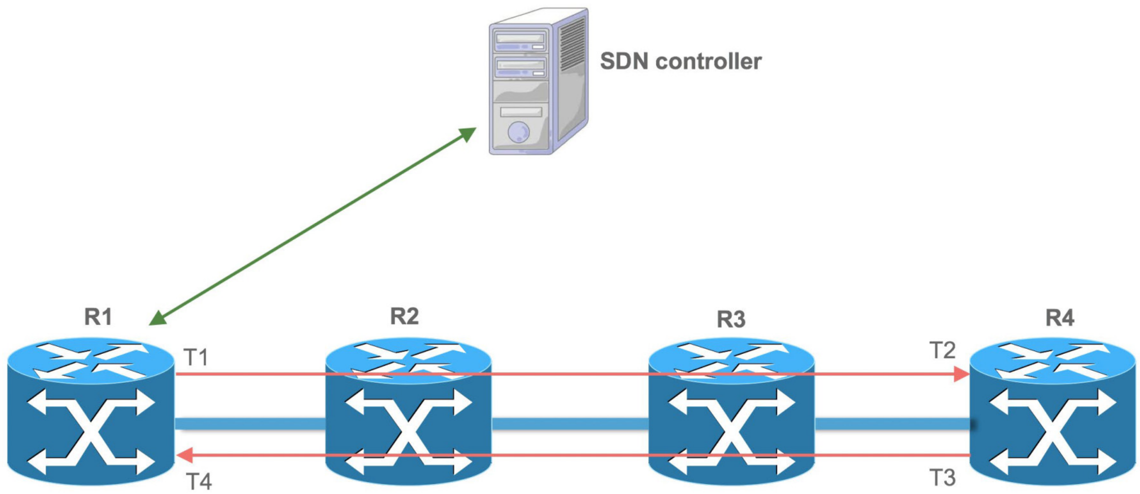

Figure 3

The third step involves three parts. For the first part of the third step, the SR policy head-end router (i.e., R1) sends a probe packet to the tail-end router (i.e., R4) as described in RFC 6374. This is similar to measuring delay of the end-to-end path between R1 and R4. However, an accurate one-way end-to-end path delay cannot be calculated only using the four timestamps (T1, T2, T3, and T4) contained in the returned probe packets as the clocks at R1 and R4 are not synchronized.

For the second part of the third step, the SR policy head-end router (i.e., R1) requests the relative clock offset of the SR policy tail-end router (i.e., R4) from the SDN controller. The SDN controller calculates the relative clock offset of R4 with respect R1 by adding up the relative clock offsets of the routers along any path between R1 and R4. For example, CO41 = CO21 + CO32 + CO43. The calculated clock offset (i.e., CO41) is then sent to R1 by the SDN controller if R1 computes the delay values.

For the third part of the third step, the R1 node or the SDN controller determines the actual one-way end-to-end path delay for the SR policy as follows:

Actual one-way end-to-end path delay = T2 - T1 - CO41

For accuracy, this method provides for the ability to accurately calculate relative clock offsets between routers at a given time. If clock frequencies of the two clocks are not equal or one clock drifts over time, the calculated relative clock offset will deviate from the actual over time. This error can be minimized by re-sending the relative clock offset to the SDN controller whenever the difference between the latest relative clock offset and the previously advertised relative clock offset exceeds a predefined threshold. Typically, the drift of a clock over a year is less than a few milliseconds.

For requirements involving this method, line card clocks (which are used for timestamping probe packets) within a router must be synchronized. Single stage forwarding routers meet this requirement by default in which a single ingress/egress forwarding plane has a single clock source. Multi-stage (e.g., 1.5 or 2 stage) forwarding routers require line cards and forwarding Application-Specific Integrated Circuits (ASICs) to synchronize clocks within the routing system; this is achievable transparently to an operator.

Distributed Solution

In the distributed solution, the head-end of a SR Policy sends a clock offset query message along the SR Policy's end-to-end path. The query message is processed at each transit node. Once a transit node receives this message, it calculates clock offset between itself and its upstream node using delay measurement probe packets. Each transit node then sends the calculated clock offsets to the head-end. After the head-end receives all clock offsets, the tail-end's clock offset with respect to the head-end is calculated.

Several advantages are realized through the methods and solutions described herein. For example, the methods provide for the ability to determine accurate one-way end-to-end path delays for SR and SRv6 TE policies without requiring clock synchronization (such as PTP) and symmetric forward and reverse end-to-end paths. Further, the methods involve no additional operational costs for network operators as the proposed method is a part of existing Performance Measurement (PM) infrastructure in routers, and the telemetry for both delay measurement and relative clock offsets can be part of the same new software module on the SDN controller. In addition, the methods do not require extra signaling or another new protocol to determine relative clock offsets. For example, routers already measure link delays, as a part of the PM feature, for path calculation. Relative clock offsets are calculated using the timestamps collected for the link delay measurements.

Accordingly, two solutions are provided herein: 1) a distributed solution without using a controller, and 2) a centralized solution using a controller. In the distributed solution, head-end of a SR Policy sends a clock offset query message along the SR Policy's end-to-end path. The query message is processed at each transit node. Once a transit node receives this message, it calculates clock offset between itself and its upstream node using delay measurement probe packets. Each transit node then sends the calculated clock offsets to the head-end. After the head-end receiving all clock offsets, tail-end's clock offset with respect to the head-end is calculated.

In the centralized solution, each node calculates clock offsets of its adjacent nodes. The calculated clock offsets are then sent to the controller for calculating clock offsets of tail-end nodes.

Each of the distributed and centralized solutions has its own advantages. A key advantage of the distributed solution is that it calculates clock offsets of the nodes that are on an end-to-end path of a SR policy only. On the other hand, centralized solution calculates clock offsets of all the nodes.

The main advantage of the centralized solution is that it calculates clock offset of a node only once, and use that clock offset for calculating end-to-end clock offsets of all the paths that traverse through that node. However, the distributed solution would require each transit node to send clock offset of the same neighbor multiple times; once for each SR policy that has an end-to-end path which goes via that transit node. In addition to that, implementation of the centralized solution would be simpler as nodes already send link delay metrics (i.e., link delay measurements) to the SDN controller for low-latency path calculation.

Interop has already been achieved for sending link delay metrics to the SDN controller, using standardized Type Length Values (TLVs) to distribute link delay metrics. Sub-TLVs can be added to these link delay metric TLVs to distribute clock offset information along with link delay metrics. Thus, sending clock offsets to the SDN controller would be an extension to the existing protocol for sending link delay measurements. Further, SDN controllers also collect delay metrics via streaming telemetry. It is easy to define an additional xpath for streaming the offset values from a router.

One of the key reasons for sending link delay measurements to an SDN controller is that the controller has the view of the network across multiple Interior Gateway Protocol (IGP) domains and Autonomous Systems (AS's). Thus, the controller is capable of calculating end-to-end paths for SR policies that spans across multiple domains. Without a controller, end-to-end paths only within an IGP domain can be calculated.

NTP/PTP and the method described herein use the same clock offset calculation. The difference between NTP/PTP and the method described herein are in the protocol. For example, the protocol of NTP/PTP synchronizes clocks of devices using the clock offset between them. However, when the links between them are asymmetric, clock synchronization is not accurate. On the other hand, protocols of the methods described herein combines clock offsets between adjacent nodes to calculate clock offset of a tail-end node with respect to a head-end node. The tail-end's clock offset is then used for calculating accurate one-way delay over asymmetric end-to-end links.

In summary, solutions are described herein that provide for one-way end-to-end path delay measurements for SR and SRv6 TE policies without clock synchronization in SDNs. A centralized solution is provided using a controller while a distributed solution is provided without using a controller. The ability to measure one-way end-to-end delay for a SR Policy is a necessity for SLA assurance by operators. Previous solutions have all required clock synchronization among nodes along the path, which is known to be complex to administer and maintain, or have measured bidirectional delay over symmetric forward and reverse paths. The proposed one-way end-to-end path delay measurement methods described herein has neither of those two requirements.

For the centralized solution, novelty of the method lies in its simplicity. In the method for the centralized solution, a SDN controller learns clock offsets among adjacent routers and then calculates end-to-end clock offset for calibrating a one-way end-to-end delay measurement. Use of an end-to-end clock offset to calibrate measured one-way end-to-end delay eliminates the requirement of synchronizing clocks. Calculation of the end-to-end clock offset by adding clock offsets of the routers on an end-to-end path eliminates the requirement of symmetric forward and reverse paths. Thus, this approach is simple, novel, and disproportionately impactful on the operational cost of measuring delay in a network.