March 2022

CONTROL PLANE CONVERGENCE MEASUREMENT IN SD-WAN FABRIC

Eugene Khabarov

CONTROL PLANE CONVERGENCE MEASUREMENT IN SD-WAN FABRIC

AUTHOR:
Eugene Khabarov

ABSTRACT

In a modern enterprise network, particularly in large-scale software-defined wide area network (SD-WAN) overlays, there is a constant need to measure convergence, also known as route propagation time, to ensure that a network may react quickly to failures and steer traffic over alternative or backup paths in such a manner that business applications and users experience minimal disruption. To address such a need, techniques are presented herein that support a network-wide technique to automatically and granularly measure control plane convergence time across a SD-WAN fabric, without manual command-line interface (CLI) intervention on a router-by-router basis, and report the results using a "single pane of glass" (e.g., a management controller or a network management system (NMS)). Unlike other solutions, the presented techniques do not rely on probing of any kind in a data plane, thus obviating the consumption of additional bandwidth or router central processing unit (CPU) utilization (which would arise under the responses that are necessary to probes).

DETAILED DESCRIPTION

In a modern enterprise network, particularly in large-scale software-defined wide area network (SD-WAN) overlays, there is a constant need to measure convergence, also known as route propagation time. Such a measurement requirement is driven by the need to ensure that a network may react quickly to failures and steer traffic over alternative or backup paths in such a manner that business applications and users experience minimal disruption. Those business applications and users encompass, specifically, videoconferencing facilities and voice over Internet Protocol (VoIP) consumers as they have become even more important recently with staff mainly working remotely. There are many solutions for achieving redundancy, resilience, and alternative path selection itself, but thus far no good solutions exist to actually measure convergence and route propagation time, in a precise fashion, in order to estimate the impact of those technologies on a network.

As one example, an escalation team at a large network equipment vendor reported that in an SD-WAN overlay of a large retail client there was a constant routing churn and wide area network (WAN) link instability on multiple branch routers that was causing controllers to operate at 100% of central processing unit (CPU) load due to a constant recalculation for alternative best paths through the overlay (i.e., a best path calculation). Before a best path calculation process is completed, a controller cannot advertise routing information to any routers in the networks resulting in huge delays in route propagation times. This was particularly concerning for the cases when new branch routers were deployed. Once the escalation team was involved and the problems causing the route propagation times were identified, yet another problem surfaced – that is, how to measure the route propagation time (i.e., convergence) to estimate the impact of all of the different improvements, including software enhancements and proposed configuration changes, in such a huge overlay with three thousand routers?

Currently, all such measurements are performed manually by network engineers with the help of tools like `ping`, `traceroute`, and `mtr`. Those types of tools rely on so-called "probes" like Internet Control Message Protocol (ICMP) echo packets. As with any manual measurements, the obtained results tend to be inaccurate. Additionally, it is not possible to scale such a technique. Even automation (with, for example, the help of scripting) cannot help at scale because it still must be enabled on a router-by-router basis (rather than for an entire overlay). Attempts have been made to improve the situation with route convergence measurement, but such attempts take a one "box" approach.

To address the types of challenges that were described above, techniques are presented herein that support a technique for automatically measuring control plane convergence time across an SD-WAN fabric without manual command-line interface (CLI) intervention on a router-by-router basis. The presented techniques include, among other things, support for an on-demand method that enables the measurement of fabric-wide control plane convergence using a "single pane of glass."

Aspects of the presented techniques may be used to measure the overlay control plane health for all or some specific network prefixes (i.e., subnets) and then report route propagation time results, either overall or on a per-fabric element basis, through a "single pane of glass" on a management controller or a network management system (NMS).

Unlike other approaches, the presented techniques do not concentrate on one particular router. Rather, they support a holistic, network-wide approach for the measurement of route propagation time (i.e., control plane convergence) across an entire enterprise network (e.g., an SD-WAN fabric). Moreover, the presented techniques may be used to the measure route propagation time for a specific prefix through an overlay rather than concentrating on just a specific router or line card convergence. This is because the final result of a whole network being in a fully-converged state is more important than the convergence of any single router while the rest of the network may still not be in a converged or ready state. Equally, convergence measurements for one particular critical prefix, or a set of specific critical prefixes, may be more important than the overall network convergence. The same can be said about troubleshooting and investigation tasks that concentrate on specific sources, destinations, or prefixes.

Unlike other solutions, the techniques presented herein do not rely on probing of any kind in a data plane. As a result, they do not result in the consumption of additional bandwidth or router central processing unit (CPU) utilization (which would arise under the responses that are necessary to probes).

Figure 1, below, depicts elements of an exemplary overall flow according to aspects of the techniques presented herein and reflective of the narrative that was presented above.

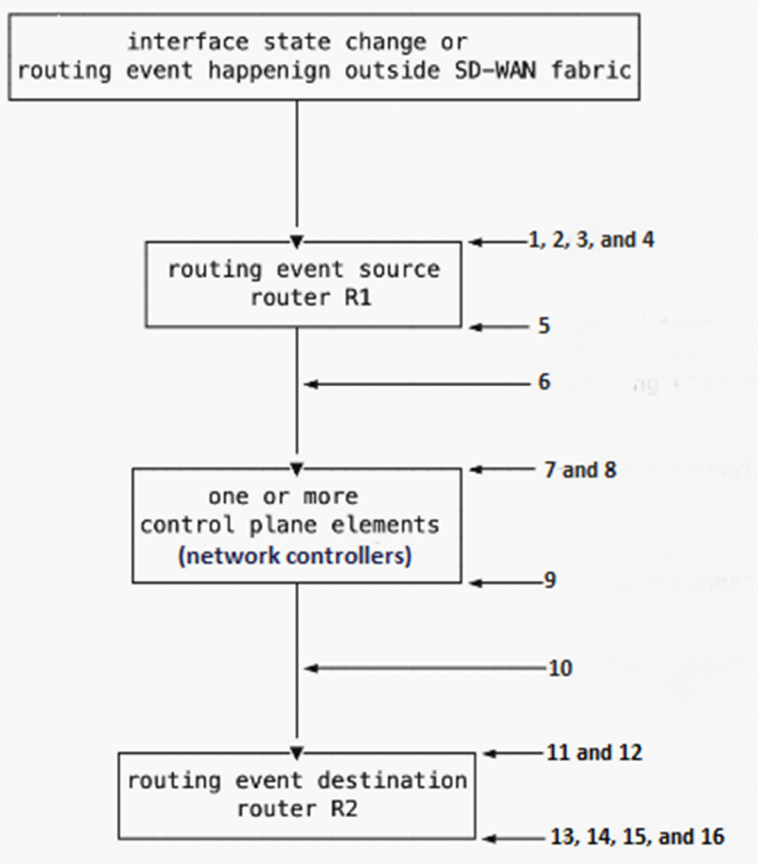

Figure 1: Exemplary Overall Flow

The exemplary flow that is depicted in Figure 1, above, identifies a series of steps. Those steps, which are labeled 1 through 16 in the figure, will be briefly described and then described in more detail later in the below narrative.

Steps 1, 2, 3, and 4 encompass recording a timestamp for (respectively) the receipt of a routing update, the installation of or change to a route in a routing information base (RIB), the installation of or change to a route in a forwarding information base (FIB), and the placement into an RIB-in of an overlay routing protocol.

Step 5 includes recording a timestamp for the completion of a best-path calculation and a route installation into an RIB-out while Step 6 encompasses sending an update containing timestamps that are encoded into an update packet and optionally an event source router identifier.

Step 7 encompasses the recording a timestamp on the receipt of a routing update, Step 8 includes the placement of same into an RIB-in of an overlay routing protocol, and

under Step 9 a timestamp is recorded for the completion of a best-path calculation (e.g., the installation of a route into an RIB-out for a destination router).

Step 10 encompasses sending an update containing timestamps that are encoded into an update packet and optionally processing a controller identifier.

Step 11 includes the recording of a timestamp on the receipt of a routing update and Step 12 encompasses the placement of same into an RIB-in of an overlay routing protocol.

Step 13 includes the recording of a timestamp for the completion of a best-path calculation, under Steps 14 and 15 a route is installed into a RIB and a FIB (respectively), and Step 16 encompasses optionally recording a receiving router identifier and exporting results through telemetry to a management plane controller or NMS.

Aspects of the techniques presented herein encompass a management controller or NMS (i.e., a "single pane of glass") that allow a network engineer to enable measurements – for a specific prefix, for a set of prefixes, or for prefixes that are coming from particular sources and that are going to particular destinations – based on currently-required troubleshooting or health checking tasks. Additionally, aspects of the presented techniques support the reporting of results and act as a destination for telemetry data for reporting. A management controller may be responsible for enabling measurements for a specific prefix or specific prefixes across the fabric on one or more sources, receivers, and control plane elements through a management protocol such as `netconf` or `restconf`.

Referring back to Figure 1, above, a routing event source router (RESR) R1 represents the edge of an SD-WAN fabric that acts as a source of a control plane or routing event (such as an interface state change) or that propagates routing updates coming from outside of the fabric (e.g., due to changes outside of the fabric where routers may not be capable of the measurement techniques as prescribed by the techniques presented herein). R1 records timestamps in different logs in, for example, a binary tracing format for efficiency. R1 may or may not have the ability to perform a granular check on the timestamps for Steps 1, 2, or 3. For example, on some routers conditional Internet Protocol (IP) routing and scalable switching debugging may be used to measure the times and record timestamps in binary trace files. It is important to note that Step 4 is a mandatory step that relies on SD-WAN overlay routing protocol (ORP) attributes. For example, a router may employ Cisco's SD-WAN overlay management protocol (i.e., an OMP) with additional

attributes to encode a timestamp when a route is placed into an OMP RIB-in. Once a best-path calculation is completed, R1 must record a timestamp into the ORP attribute of a particular prefix before placing it into the RIB-out of an ORP (as depicted in Step 5).

Under Step 6, R1 sends an update to the fabric control plane elements (e.g., a network controller) and optionally encodes its identifier into ORP attributes. R1 may also record events and corresponding timestamps into different locally-stored logs (preferably in a binary format for faster processing and more compact storage)

Again referring back to Figure 1, above, control plane elements (e.g., a network controller) disseminates control plane information across a fabric. Such entities must record timestamps before placing a route into an ORP RIB-in (as depicted in Step 8) and after a best-path calculation before placing a route into an ORP RIB-out (as depicted in Step 9). Such entities must also send updates, with timestamps that are encoded into prefix attributes, to the receiver of a routing update (as depicted in Step 10). Such entities may optionally encode an identifier of a control plane element that has processed an update into ORP prefix attributes and may also record events and corresponding timestamps into locally-stored logs (preferably in a binary format for faster processing and more compact storage).

Once again referring back to Figure 1, above, a routing event destination router (REDR) R2 represents the other edge of an SD-WAN fabric that acts as an ultimate receiver of a control plane or routing event. A control plane is converged for a specific measured prefix only when a route is installed into the RIB and/or the FIB on a REDR. Such a router records timestamps (upon receipt of a routing update and placement into an ORP RIB-in) in the logs (recording them in, for example, a binary tracing format for efficiency) and then encodes them as an attribute of an ORP (as depicted in Step 13). Just like R1, R2 may or may not have the ability to perform a granular check on the timestamps for Steps 14 or 15. A final action of R2 is to decode information that is stored in the attributes of an ORP about timestamps and the identifiers of routers and control plane elements that are involved in the network prefix processing and export same to a management plane controller or NMS through a telemetry protocol in a most suitable format.

It is important to note that while elements of the Border Gateway Protocol (BGP) address a portion of the challenges that were described above (through, for example, the

use of an external emulator such as a testing, visibility, etc. traffic generator), the techniques presented herein support a mechanism that, as described and illustrated above, enables the inline measurement of control plane convergence within an SD-WAN fabric without the use of any external emulator. Further, a centralized management controller cannot be considered equal to an external emulator because when, for example, a SD-WAN controller is not available the presented techniques will still work and data may be gathered from a receiving router through a CLI.

Aspects of the presented techniques allow for a per-node convergence measurement (e.g., how much time it takes to process a route through a sender, controller(s), and receivers) and hence may be used for the day-to-day troubleshooting of SD-WAN overlay networks unlike the process that is described in the Internet Engineering Task Force (IETF) Request for Comments (RFC) 7747 where an emulator measures overall convergence and only for connected routers.

Further, aspects of the presented techniques may be voluntarily enabled network-wide (e.g., for an entire overlay) for a specific prefix (e.g., per prefix) or for a specific set of routers or controllers.

While RFC 7747, which was noted above, discusses a particular traffic requirement (see, for example, Sections 5.1.5 or 5.2.1), aspects of the techniques presented herein support a mechanism that introduces no additional traffic load. Under the presented techniques only control plane packets are involved and required for measurements, thus reducing any disturbances to an overlay that is undergoing testing and improving the accuracy of the measurements.

Moreover, in SD-WAN overlays the traffic usually flows through different paths (between routers directly) compared to the control plane traffic (which travels through one or more controllers) and hence the methodology that is described in RFC7747 cannot be applied in SD-WAN overlays.

It is also important to note that aspects of the techniques presented herein require that all of the included devices must be synchronized through, for example, the Network Time Protocol (NTP) to support accurate measurements.

In summary, techniques have been presented herein that support a network-wide technique to automatically and granularly measure control plane convergence time across

a SD-WAN fabric, without manual CLI intervention on a router-by-router basis, and report the results using a "single pane of glass" (e.g., a management controller or an NMS). Unlike other solutions, the presented techniques do not rely on probing of any kind in a data plane, thus obviating the consumption of additional bandwidth or router CPU utilization (which would arise under the responses that are necessary to probes).