Network Working Group
Internet-Draft
Intended status: Informational
Expires: November 8, 2019

M. Chen
X. Geng
Huawei
Z. Li
China Mobile

May 7, 2019

Segment Routing (SR) Based Bounded Latency
draft-chen-detnet-sr-based-bounded-latency-01

Abstract

One of the goals of DetNet is to provide bounded end-to-end latency for critical flows. This document defines how to leverage Segment Routing (SR) to implement bounded latency. Specifically, the SR Identifier (SID) is used to specify transmission time (cycles) of a packet. When forwarding devices along the path follow the instructions carried in the packet, the bounded latency is achieved. This is called Cycle Specified Queuing and Forwarding (CSQF) in this document.

Since SR is a source routing technology, no per-flow state is maintained at intermediate and egress nodes, SR-based CSQF naturally supports flow aggregation that is deemed to be a key capability to allow DetNet to scale to large networks.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on November 8, 2019.

Table of Contents

1. Introduction

Deterministic Networking (DetNet) [I-D.ietf-detnet-architecture] is defined to provide end-to-end bounded latency and extremely low packet loss rates for critical flows. For a specific path, the end-to-end latency consists of two parts: 1) the accumulated latency on the wire, 2) the accumulated latency of nodes along the path. The former can be considered as constant once the path has been determined. The latter is contributed by the latency within each node along the path. So, to guarantee the end-to-end bounded latency, control the bounded latency within a node is the key. If

every node along the path can guarantee bounded latency, then end-to-end bounded latency can be achieved.

[I-D.finn-detnet-bounded-latency] gives a framework that describes how bounded latency and zero congestion loss are achieved. It introduces a parameterized timing model that can be used by DetNet solutions by selecting a corresponding Quality of Service (QoS) algorithm and resource reservation algorithm to achieve the bounded latency and zero congestion loss goal.

This document defines how to leverage Segment Routing (SR) [RFC8402] to implement bounded latency, which is called Time Aware Segment Routing(TA-SR). A segment is associated with a topological instruction, which instruct a node to forward the packet via a specific outgoing interface, as it is defined in [RFC8402]. At the same time, the segment is also associated with DetNet bounded latency service. Specifically, the segment ID(SID) is used to carry and specify the "sending time" of a packet, and some mechanisms can be used to ensure that the packet will be transmitted in that specified period of sending time, which is called Time Aware Segment Routing(TA-SR).

The TA-SR architecture supports any type of control plane: distributed (IS-IS or OSPF or BGP), centralized (NETCONF or PCEP or BGP), or hybrid (PCEP or BGP).

The TA-SR architecture can be instantiated on various data planes, including TA-SR over MPLS (TA-SR MPLS) or TA-SR over IPv6 (TA-SRv6).

2. Terminology

All the terminologies used in this document are extensions of [RFC8402].

Time Aware Segment:

Time Aware SID:

TA-SR MPLS SID:

TA-SRv6 SID:

TA-SR Domain:

TA-SR Globle Block (SRGB):

TA-SR Local Block (SRGB):

TA-Adjacency Segment:

Forwarding Time Base: besides: the node uses the SID as an entry to get the Egress interface with Forwarding Information Base(FIB); Similarl, the node can use the SID as an entry to get the sending time of the packet, with Forwarding Time Base.

3. Cycle Specified Queuing and Forwarding

3.1. CSQF Basic Concepts

By specifying the sending cycle of a packet at a node and making sure that the packet will be transmitted in that cycle, CSQF can achieve bounded latency within the node. By specifying the sending cycle at every node along a path, the end-to-end bounded latency can be achieved.

To support CSQF, similar to Cyclic Queuing and Forwarding (CQF) [IEEE802.1Qch], the sending time of an output interface of a node is divided into a series of equal time intervals with the duration of T. Each time interval is called a "cycle", and each cycle corresponds to a queue. During a cycle, only the corresponding queue is open and all the packets in that queue will be transmitted. CSQF can not only control the bounded latency at every node along a path, but regulate the traffic at each node as planned. Therefore, no congestion will occur.

Figure 1 provides an overview of CSQF.

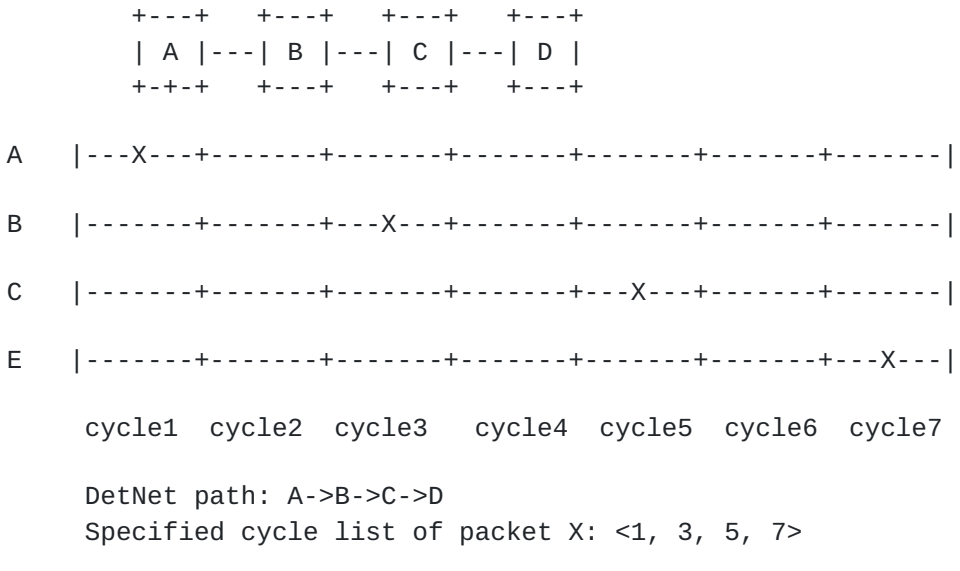


Figure 1: CSQF Overview

CSQF has the following characteristics:

- The sending time (cycle) of a packet at each node along a path is specified so that the packet will be transmitted in the specified cycles, hence to guarantee the end-to-end bounded latency.

- The specified cycles are calculated by fully considering the link delay, processing delay and the available cycle resources, resulting in no bandwidth waste and no congestion (cycle-based traffic regulation).

- Segment routing (SR) is used. Specifically, a SID is used to indicate in which cycle and to which output interface that a packet is specified to transmit, and an SR SID list is used to carry the specified cycles along a path. With SR, there is no per-flow states maintained at the intermediate and egress node. As a result, scalability is greatly improved compared to a solution that maintains flow state at each hop.

- Flow aggregation is naturally supported by introducing SR and cycle-based scheduling.

3.2. CSQF Queuing Model

In Cyclic Queuing and Forwarding (CQF) [IEEE802.1Qch], time is divided into numbered time intervals, and each time interval is called a cycle; the critical traffic is then transmitted and queued for transmission along a path in a cyclic manner. With CQF, the delays experienced by a given packet are as follows:

- The maximum end-to-end delay = (N+1) * T;

- The minimum end-to-end delay = (N-1) * T;

- Where the N is the number of hops and T is the duration of the cycle.

CQF assumes that a packet is transmitted from an upstream node in a cycle and the packet must be received at the downstream node in the same cycle, and it must be transmitted in the next cycle to the nexthop node. This assumption leads to very low bandwidth utilization when the link delay, processing delay, etc., factors cannot be considered as trivial. To guarantee this assumption, more bandwidth has to be reserved as a guard band for each cycle, and the effective bandwidth for DetNet service will be greatly reduced.

CSQF improves on CQF by explicitly specifying the sending cycles at every node along the path. This relieves the limitation that the

sending (at the upstream node) and receiving (at the downstream node) have to be in the same cycle. For CSQF, the cycle to use depends on traffic planning and path calculation. The path calculation will consider the available cycle resources, bandwidth, and delay constraints.

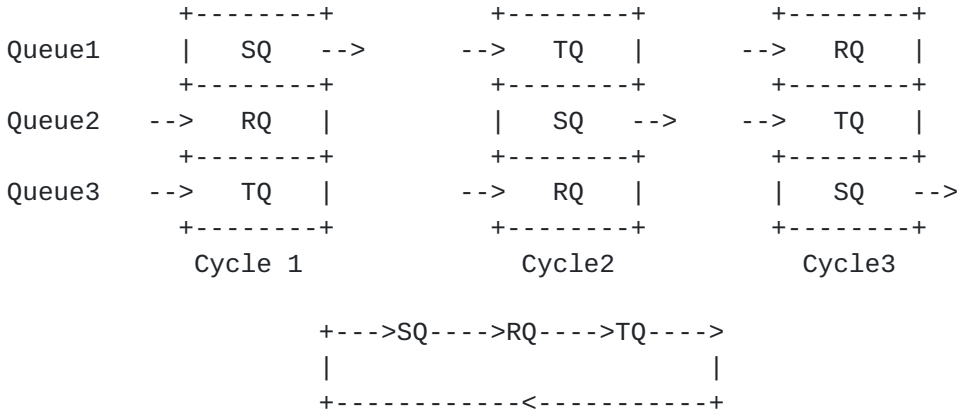


Figure 2: CSQF Queuing Model

For CSQF, three queues (in theory, two or more queues work as well) for each output interface are used. During a particular cycle, only one queue is open and the packets in that queue will be transmitted. This queue is called the sending queue (SQ). The other two queues are closed and can enqueue packets. One of them is called the receiving queue (RQ). The third queue is called the tolerating queue (TQ).

The RQ is used for receiving the packets that are expected to be transmitted in the next cycle. The TQ is used for tolerating the packets that come a bit early due to processing delay variation (processing jitter) or other reasons (e.g., packets are not transmitted as required by the traffic specification). Both RQ and TQ can have the capability to absorb a certain amount of processing jitter and traffic bursts. The upper bound of the absorbing capacity is 2T. In order to increase the jitter/burst absorbing capacity, a four or more-queue model can be used. If the processing delay and traffic bursts are small, two-queue model works as well.

The roles of the three queues are not fixed, and on the contrary, they rotate with each cycle change. As showed in Figure 2, during cycle 1, queue 1 is SQ, queue 2 is RG and queue 3 is TQ; during cycle 2, queue 1 is TQ, queue 2 is SQ and queue 3 is RQ, during cycle 3, queue 1 is RQ, queue 2 is TQ and queue 3 is SQ. That means, for a particular queue, its role will rotate as "...->SQ->RQ->TQ->SQ->...", the starting role of a queue can be any one of the three roles.

In CSQF, a cycle corresponds to a queue. There are several ways to do cycle to queue mapping. The simplest mapping between cycles and queues is 1:1 mapping. There could be N:1 mapping, but that requires more identifiers, which in the case of segment routing, would require more SIDs. This document does not specify which mapping should be used. The mapping choice is left to the operator.

3.3. CSQF Timing Model

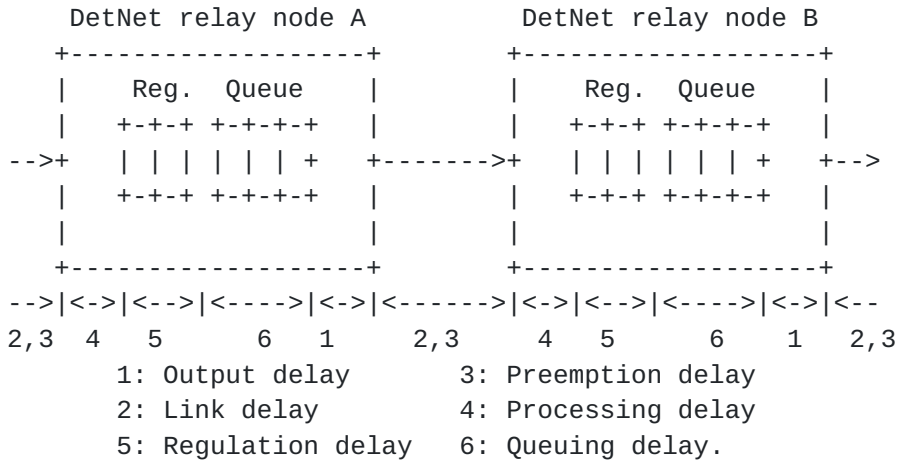


Figure 3: Timing model for DetNet

The DetNet timing model in Figure 3 is defined in [I-D.finn-detnet-bounded-latency]. It details the delays that a packet can experience from hop to hop. There are six delays, the detailed explanation of which can be found in [I-D.finn-detnet-bounded-latency]. This document simplifies the above model as follows:

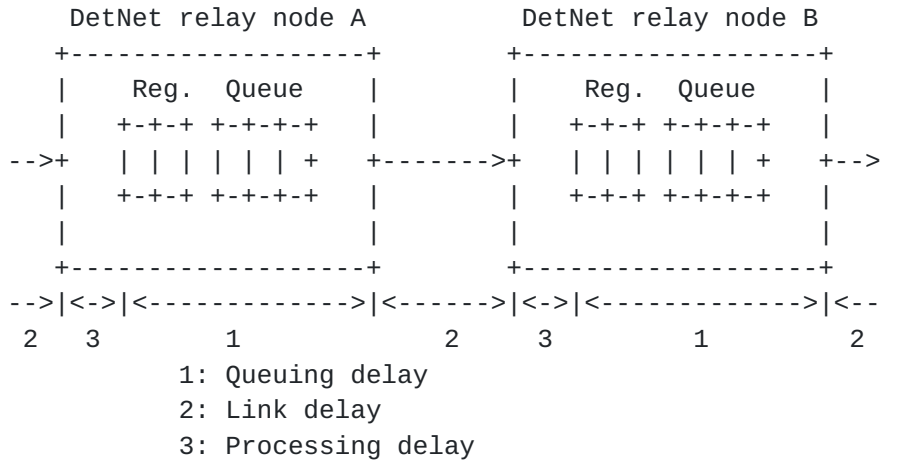


Figure 4: Simplified Timing model for DetNet

In this simplified timing model, only three delays are defined. The queuing delay in this new model includes the output delay, regulation delay, and queuing delay that are defined in the DetNet timing model (Figure 3). The link delay defined in this document includes the link delay and the preemption delay defined in [I-D.finn-detnet-bounded-latency]. The processing delay is the same as defined in [I-D.finn-detnet-bounded-latency].

To further simplify the model, it assumes that the link delay only depends on the distance of the link. Once the DetNet path has been determined, the link delay can be considered as constant. The processing delay and queuing delay are variable but have their upper bounds.

For the processing delay, there are two bounds: minimum processing delay (Min-P-Delay) and maximum processing delay (Max-P-Delay).

- Thus, the maximum processing jitter (Max-P-Jitter) = Max-P-Delay - Min-P-Delay.

As described in Section 2.2, both the RQ and TQ can be used for absorbing processing jitter, and the upper bound of the absorbing capacity is 2T. So, if the processing jitter is less than 2T, the three-queue model can work. Otherwise, more buffer is needed to absorb the jitter, through increasing the duration of the cycle or by adding more queues. Increasing the duration of the cycles is equivalent to increasing the depth of the queues (adding more buffer for each queue).

With above, for CSQF, the delays experienced by a given packet are as follows:

- The maximum end-to-end delay = Link delay + N * (Max-P-Delay + 2T);

- The maximum end-to-end jitter = 2T;

- Where N is the number of hops and T is the duration of a cycle.

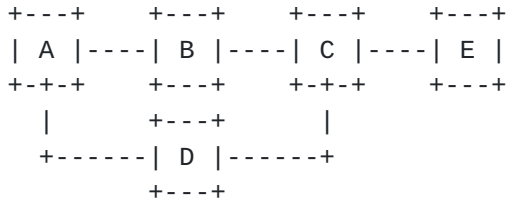
3.4. Congestion Protection and Resource Reservation

Congestion protection is the key for bounded latency and zero congestion loss. An essential component of DetNet is Traffic Engineering (TE), so that dedicated resources can be reserved for the exclusive use of DetNet flows. To avoid congestion, two or more flows must be prevented from contending for the same resource. For normal TE, the critical resource is bandwidth, but in the case of CSQF, the critical resource is interface occupation time. Bandwidth

is an average value, which can generally guarantee the quality of service generally, but bursts and congestion may still occur. By comparison, the interface occupation time is an absolute value, which can avoid packet packets conflicting for the same resource by controller computation and time allocation for different flows. The unit of time allocation is the cycle, and a Traffic Specification, the flow transmission description, is necessary for the computation.

CSQF uses segment routing SIDs to carry the time allocation information (the cycle), and it ensures that a node can schedule different packets without conflict and forward the packets at the proper time. The resource reservation is not explicitly implemented by a control plane protocol, such as Resource Reservation Protocol - Traffic Engineering (RSVP-TE) or Stream Reservation Protocol (SRP). Rather, it is guaranteed by the SR controller, which maintains the status of different flows and time occupation of all the network devices in the domain. This is called the Virtual Resource Reservation (VRR) in this document.

3.5. An Example of CSQF




```
A   |---X---+-------+-------+-------+-------+-------+-------|

B   |-------+-------+---X---+-------+-------+-------+-------|

C   |-------+-------+-------+-------+---X---+-------+-------|

E   |-------+-------+-------+-------+-------+-------+---X---|

     cycle1  cycle2  cycle3   cycle4  cycle5  cycle6  cycle7

     DetNet path: A->B->C->E
     Specified cycle list <1, 3, 5, 7>
```

Figure 5: CSQF Example

As showed in Figure 5, there is a DetNet path (A->B->C->E), and a packet (X) is expected to be transmitted in cycle 1 at node A, in cycle 3 at node B, in cycle 5 at node C and in cycle 7 at node E. A

cycle list <1, 3, 5, 7> is attached to the packet, and the packet will be transmitted along the path as the specified cycles.

Given the topology as above, assume the duration of a cycle is 10us; the link delays between nodes are the same (e.g., 100us); the minimum processing delay at each node = 10us, the maximum processing delay at each node is 20us, so the maximum processing jitter is 10us.

For a given packet that is transmitted along the path(A->B->C->D->E), the experienced maximum end-to-end delay is:

(N-1) * link delay + N * (maximum processing delay + 2T)

= 3*100 + 4* 40

= 460 (us)

The maximum end-to-end jitter is always 2T (20us).

4. Segment Routing Extensions for CSQF

This document defines a new segment that is called a Cycle Segment, which is used to identify a cycle. A Cycle Segment is a local segment and is allocated from the Segment Routing Local Block (SRLB)[RFC8402].

A Cycle Segment has two meanings: 1) identify an interface/link, just like the adjacency segment does; 2) identify a cycle of the interface/link. To specify to which interface and in which cycle a packet should be transmitted, it just needs to attach a Cycle Segment to the packet. By attaching a list of Cycle Segments to a packet, it can not only implement the explicit route of the packet that is required by DetNet [I-D.ietf-detnet-architecture], but also specify the sending cycle at each node along the path without maintaining per-flow states at the intermediate and egress nodes. Hence, it naturally supports flow aggregation, and that allows DetNet to support large number of DetNet flows and scale to large networks.

Normally, several SR SIDs are required to be allocated for each CSQF capable interface. How many SIDs are allocated depends on how many cycles are used. Given a three-queue model and a 1:1 cycle to queue mapping is used, three SIDs will be allocated for each CSQF capable interface. For example, given node A, SR-MPLS SIDs 1001, 1002, and 1003 are allocated to one of its interfaces. SID 1001 identifies cycle 1, SID 1002 identifies cycle 2, SID 1003 identifies cycle 3.

The SR [RFC8402] can be instantiated on various data planes. There are two data-plane instantiations of SR: SR over MPLS (SR-MPLS) and

SR over IPv6 (SRv6). Both SR-MPLS and SRv6 SIDs can be used for CSQF cycle identification. The mapping (IGP extensions) between a cycle and a SID will be defined in a separate document.

4.1. Time Aware Adjacency Segment(TA-Adj-SID)

An Time Aware Adjacency segment is an IGP segment attached to a specified sending time of a unidirectional adjacency, which inheriting all the definitions of Adjacency segment defined in [RFC8402], adding new capability:

When a node binds a group of AT-Adj-SIDs V1-Vn to a local data-link L, the node MUST install the following FIB entry:

Incoming Active Segment: V1-Vn

Ingress Operation: NEXT

Egress Interface: L

When a node binds an TA-Adj-SID V1 to sending time: Cycle 1, the node MUST install the following Forwarding Time Base (FTB) entry:

Incoming Active Segment: V1

Sending Time: Cycle 1

Output Queue: Queue 1

So a packet with TA-Adj-SID V1 will be transmitted go through output queue 1 of egress interface L within cycle 1.

5. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

6. Security Considerations

7. Acknowledgements

The authors would like to thank Andrew G. Malis, Norman Finn for his review, suggestion and comments to this document.

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

8.2. Informative References

[I-D.finn-detnet-bounded-latency] Finn, N., Boudec, J., Mohammadpour, E., Zhang, J., Varga, B., and J. Farkas, "DetNet Bounded Latency", draft-finn-detnet-bounded-latency-03 (work in progress), March 2019.

[I-D.geng-detnet-conf-yang] Geng, X., Chen, M., Li, Z., and R. Rahman, "DetNet Configuration YANG Model", draft-geng-detnet-conf-yang-06 (work in progress), October 2018.

[I-D.geng-detnet-info-distribution] Geng, X., Chen, M., and Z. Li, "IGP-TE Extensions for DetNet Information Distribution", draft-geng-detnet-info-distribution-03 (work in progress), October 2018.

[I-D.ietf-detnet-architecture] Finn, N., Thubert, P., Varga, B., and J. Farkas, "Deterministic Networking Architecture", draft-ietf-detnet-architecture-12 (work in progress), March 2019.

[IEEE802.1Qch] IEEE, "IEEE, "Cyclic Queuing and Forwarding (IEEE Draft P802.1Qch)", 2017, <http://www.ieee802.org/1/files/private/ch-drafts/>.", 2016.

[RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <https://www.rfc-editor.org/info/rfc8402>.

Authors' Addresses

Mach(Guoyi) Chen
Huawei

Email: mach.chen@huawei.com

Xuesong Geng
Huawei

Email: gengxuesong@huawei.com

Zhenqiang Li
China Mobile

Email: lizhenqiang@chinamobile.com