Requirements for Scaling Deterministic Network

draft-ietf-detnet-scaling-requirements-04

Peng Liu liupengyjy@chinamobile.com
Yizhou Li liyizhou@huawei.com
Toerless Eckert tte@cs.fau.de
Xiong Quan xiong.quan@zte.com.cn
Jeong-dong Ryoo ryoo@etri.re.kr
Shiyin Zhu zhushiyin@h3c.com
Xuesong Geng gengxuesong@huawei.com

DetNet WG - IETF 118, Prague – Nov 2023

Motivations and Status

Motivations

Aiming at scaling deterministic network with large variation in latency among hops, great number of flows and/or multiple domains without the same time source, this document describes the technical requirements including the data plane enhancement requirements when the different deterministic levels of applications co-exist and are transported.

Status

This document was updated from v03 to v04 according to the discussion in the bi-weekly interim.

Add some text to enhance the explanation of requirements.

Rename some sections without adding new requirement or changing the structure.

Main Updates(v03-v04)

- Rename Section 3.1.3, Section 3.4 and Section 3.8.1
- Section 3.6. Make it clear to ‘DetNet flows’ rather than all the network flows.
- Section 3.7. Add the text related to consider end-to-end jitter bounds, achievable end-to-end latency bounds and complexity in evaluation and selection of scalable DetNet queuing mechanisms.
- Section 3.8. Change the name of the different kinds of flows in figure
- Section 4. Add the references to I-D.ietf-detnet-controller-plane-framework

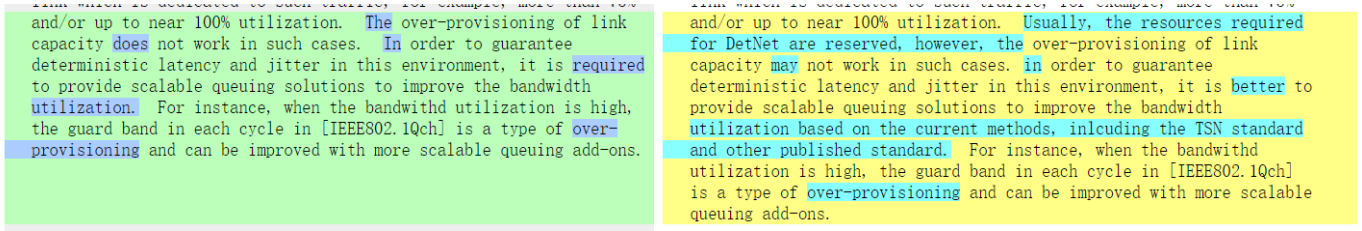
https://author-tools.ietf.org/iddiff?url1=draft-ietf-detnet-scaling-requirements-03&url2=draft-ietf-detnet-scaling-requirements-04&difftype=--html

Technical Requirements

- Req 1. Tolerate Time Asynchrony
  - Support Asynchronous Clocks Across Domains
  - Tolerate Clock Jitter & Wander within a Clock Synchronous Domain
  - Provide Mechanisms not Requiring ~~Full~~ Strict Time Synchronization
  - Provide Mechanisms not Requiring Synchronization
- Req 2. Support Large Single-hop Propagation Latency
- Req 3. Accommodate the Higher Link Speed-
- Req 4. Be Scalable to The Large Number of Flows and Tolerate High Utilization of Bandwidth
- Req 5. Tolerate Failures of Links or Nodes and TopologyChanges
- Req 6. Prevent Flow Fluctuation
- Req 7. Be Scalable to a Large Number of Hops with Complex Topology
- Req 8. Support Multi-Mechanisms in Single Domain and Multi-Domains
  - Support ~~Configuration~~ Provisioning of Multiple Mechanisms
  - Support Mechanisms Switchover Crossing Multi-domains

Req 4. Be Scalable to The Large Number of Flows and Tolerate High Utilization of Bandwidth

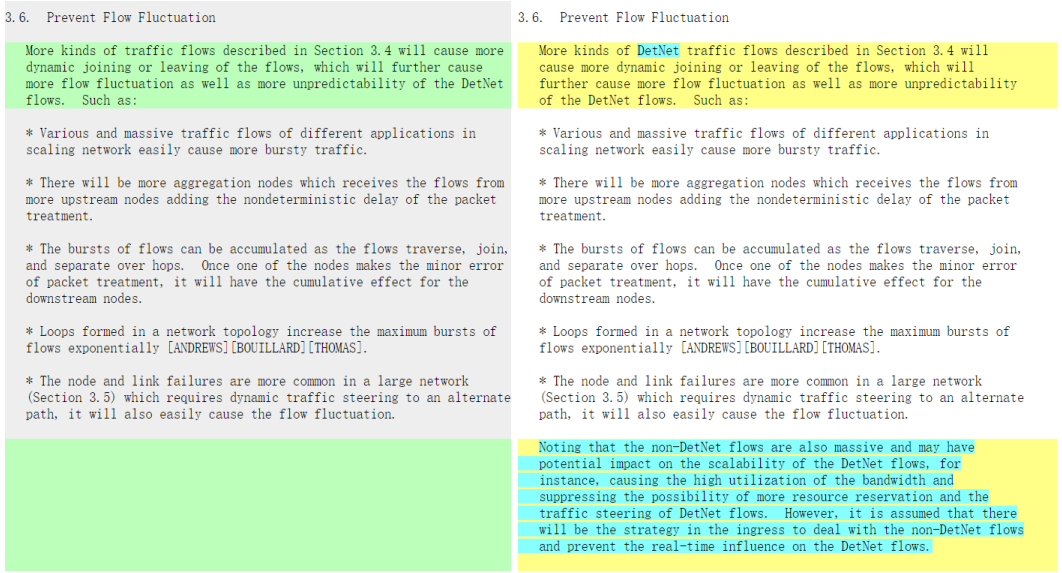
- Try to make it clear of the reqs of High Utilization of Bandwidth'. It is for the case that the bandwidth utilization is more than 75% and/or up to near 100%.
- Considering the resources required for detnets are usually reserved and it is not sure if there is any space for the queuing solutions can improve the bandwidth utilization based on the current methods except CQF, so make it optional.



Req 6. Prevent Flow Fluctuation

Add a paragraph at the end :

Noting that the non-DetNet flows are also massive and may have potential impact on the scalability of the DetNet flows, for instance, causing the high utilization of the bandwidth and suppressing the possibility of more resource reservation and the traffic steering of DetNet flows. However, it is assumed that there will be the strategy in the ingress to deal with the non-DetNet flows and prevent the real-time influence on the DetNet flows.



3.6. Prevent Flow Fluctuation

More kinds of traffic flows described in Section 3.4 will cause more dynamic joining or leaving of the flows, which will further cause more flow fluctuation as well as more unpredictability of the DetNet flows. Such as:

* Various and massive traffic flows of different applications in scaling network easily cause more bursty traffic.

* There will be more aggregation nodes which receives the flows from more upstream nodes adding the nondeterministic delay of the packet treatment.

* The bursts of flows can be accumulated as the flows traverse, join, and separate over hops. Once one of the nodes makes the minor error of packet treatment, it will have the cumulative effect for the downstream nodes.

* Loops formed in a network topology increase the maximum bursts of flows exponentially [ANDREWS][BOUILLARD][THOMAS].

* The node and link failures are more common in a large network (Section 3.5) which requires dynamic traffic steering to an alternate path, it will also easily cause the flow fluctuation.

3.6. Prevent Flow Fluctuation

More kinds of DetNet traffic flows described in Section 3.4 will cause more dynamic joining or leaving of the flows, which will further cause more flow fluctuation as well as more unpredictability of the DetNet flows. Such as:

* Various and massive traffic flows of different applications in scaling network easily cause more bursty traffic.

* There will be more aggregation nodes which receives the flows from more upstream nodes adding the nondeterministic delay of the packet treatment.

* The bursts of flows can be accumulated as the flows traverse, join, and separate over hops. Once one of the nodes makes the minor error of packet treatment, it will have the cumulative effect for the downstream nodes.

* Loops formed in a network topology increase the maximum bursts of flows exponentially [ANDREWS][BOUILLARD][THOMAS].

* The node and link failures are more common in a large network (Section 3.5) which requires dynamic traffic steering to an alternate path, it will also easily cause the flow fluctuation.

Noting that the non-DetNet flows are also massive and may have potential impact on the scalability of the DetNet flows, for instance, causing the high utilization of the bandwidth and suppressing the possibility of more resource reservation and the traffic steering of DetNet flows. However, it is assumed that there will be the strategy in the ingress to deal with the non-DetNet flows and prevent the real-time influence on the DetNet flows.

Req 7. Be Scalable to a Large Number of Hops with Complex Topology

We had the problems to refine this requirements:

- Different understanding of how to evaluate the queuing methods according to this req.
- Proposed some critiera for the latency, jitter, routing calculation and the resources scheduling, which didn't reach concensus.

The current changes from v03 to v04:

Normally, bounded latency is a constant of application regardless of the number of hops. So keeping the bounded latency with the variaty of hops is required in scaling detnet, especially for the lower end-to-end latency bound. The queuing mechanisms should be enhanced, for instance, adjust the cycle time in [IEEE802.1Qch] to meet the end to end latency while considering the feasibility with the dead time.

Normally, bounded latency and jitter is a constant of application, so for the queuing mechanisms, keeping them with the variety of hops is required in scaling detnet. The queuing mechanisms should be enhanced, for instance, adjust the cycle time in [IEEE802.1Qch] to meet the end to end latency while considering the feasibility with the dead time. The performance of a queuing mechanism can be evaluated based on the E2E latency bound, the E2E jitter bound, and the execution time required for resource scheduling, which might be constant, or linear functions, or exponential functions in terms of number of flows or network diameter.

Req 7. Be Scalable to a Large Number of Hops with Complex Topology

Thanks to David for providing the text, we will add it in the v05 soon.

Scaling networks often results in situations where an end-to-end flow involves a large number of hops, e.g., 15 or more. The network topology can also be complex, including star, ring, mesh, and their combinations, and can possibly be hierarchical. It is required to support networks with such various types of topologies and large hop Counts.

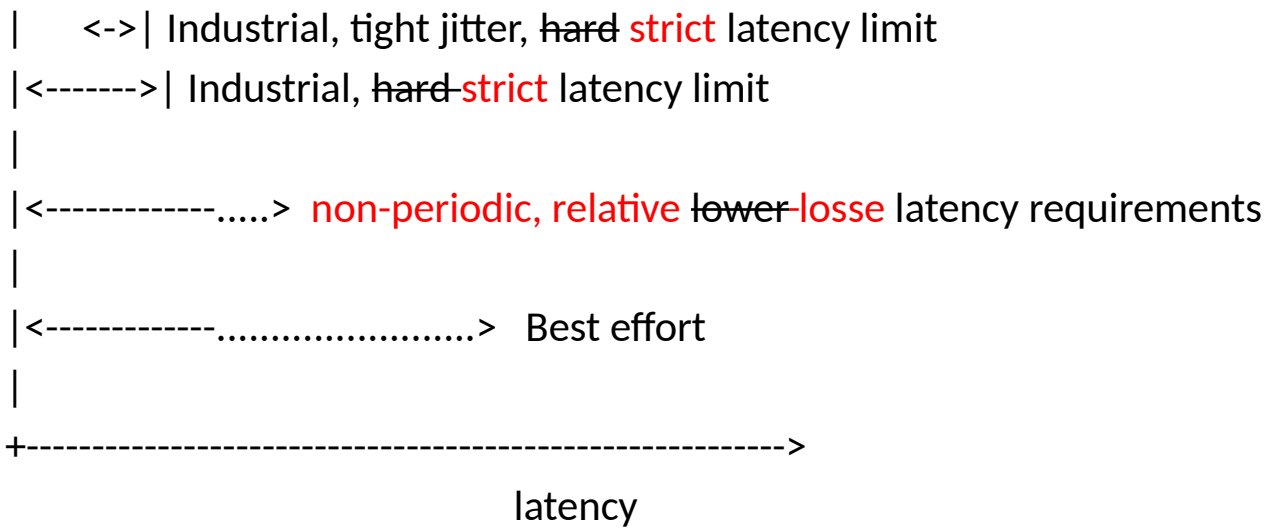
***Delivering DetNet Quality of Service (QoS) in large and complex networks requires end-to-end bounds on both latency and jitter**, as discussed in section 3.1 of [RFC8655]. Achievable end-to-end latency bounds necessarily depend on the number of hops for a flow. In the best case, the added queuing mechanism latency for **DetNet QoS is bounded by a fixed constant per hop** maximum value so that the resulting **end-to-end latency bounds are a linear function** of the number of hops in addition to the inherent forwarding latency of the nodes involved. In contrast, it is possible to achieve **fixed constant end-to-end jitter bounds** that are independent of the number of hops. Such fixed constant jitter bounds are strongly preferable to jitter bounds that increase with an increasing number of hops.*

*DetNet QoS requires resource allocation in advance (e.g., of link bandwidth and node buffer resources), as discussed in section 3.2.1 of [RFC8655]. **The complexity of resource allocation processing may range from linear** (e.g. allocating resources for each hop via a path-based resource reservation protocol such as RSVP [RFC2205]) **to potentially exponential** (e.g., if solving a complex graph optimization problem is required). This resource allocation complexity does not directly affect achievable end-to-end latency and jitter bounds, but it does surface in other areas such as the amount of computation and elapsed time required to admit a new flow to a DetNet network without disrupting the DetNet QoS provided to already admitted flows.*

Different queuing mechanisms exhibit different properties across achievable end-to-end jitter bounds, achievable end-to-end latency bounds and complexity. All three of these areas are considerations in evaluation and selection of scalable DetNet queuing mechanisms.

Req 8. Support Multi-Mechanisms in Single Domain and Multi-Domains

Critical latency requirements:



Next step

- Refine the document and address the comments