Quality of Service: Progress in IETF DetNet

Norman Finn
Huawei Technologies Co. Ltd
new-finn-detnet-QoS-progress-1118-v01

Topics

- Status of P802.1DC
- Introduction to draft-finn-detnet-bounded-latency
- Progress of draft-finn-detnet-bounded-latency
- How much of this should be in some 802.1 document?
- Future work for IEEE

Status of P802.1DC

- No new draft.

Introduction to draft-finn-detnet-bounded-latency

A personal draft has been contributed to the IETF DetNet Working Group:

draft-finn-detnet-bounded-latency

Authors: Norman Finn, Jean-Yves Le Boudec, Ehsan Mohammadpour, Jiayi Zhang, János Farkas, Balázs Varga

Abstract

This document presents a parameterized timing model for Deterministic Networking (DetNet), so that existing and future standards can achieve the DetNet quality of service features of bounded latency and zero congestion loss. It defines requirements for resource reservation protocols or servers. It calls out queuing mechanisms, defined in other documents, that can provide the DetNet quality of service.

Summary of bounded-latency draft

- An explanation of a network timing model based on Le Boudec's book, Network Calculus.
- A description of the process of creating a DetNet flow (or TSN Stream).
- A mathematical analysis of how latency and buffer space is computed – based on arrival curves and service curves, not a sum of per-hop worst-case latency.
- Descriptions of various 802.1Q queuing methods and IntServ.
- A set of parameters that can be used for reservations (TBD).

Next version of bounded-latency draft

- We will describe thoroughly the two different calculations needed:
  - The **Static** calculation: Given the complete set of flow to be accommodated by a network, their paths and bandwidth characterizations, compute the worst-case latency that can be experienced by each, and the buffer requirements in each relay node to guarantee zero congestion loss.
  - The **Dynamic** calculation : Given a network whose total capacity is limited by some set of configured parameters, and given only one flow, its path and bandwidth characterization, compute its worst-case latency and the per-relay node buffer requirements that can be guaranteed no matter what other flows may be subsequently created (subject always to the network capacity).
- We will discuss the special cases of ingress to the network, the related case of interspersed non-aware relays, and other scenarios, also.

This author's take-away from DetNet meeting

- DetNet assumes that IEEE will write standards for queuing methods.
- There is little, if anything, in bounded-latency that needs to be in a normative DetNet document.
- The document is of significant interest to DetNet, as it is the only document that tries to show that bounded latency is possible, much less how to achieve it, or what bounds can be provided.

Work for 802.1 to consider

- 802.1 uses the sum of per-hop worst-case latency to compute end-to-end worst-case latency. This should be changed, because hop-by-hop gives a very pessimistic answer.
- The analysis of Asynchronous Traffic Shaping in the bounded-latency draft belongs in IEEE Std 802.1Q, via P802.1Qcr.
  - (My understanding is that it will be, though not necessarily using the same mathematical model. **DISCUSS**.)
- 802.1Q needs to show a way to calculate the buffer space necessary to achieve zero congestion loss. (If we can't show the calculation, how does the reader know that it is possible?)

More work for 802.1 to consider: “Ingress conditioning”

- Right now, 802.1Qci specifies a byte counter that can admit a certain number of bytes per cycle. This is a defense for CQF.
- But, it is not very good for accepting a Talker that is not, itself, running CQF. Much better would be to specify a something like a burst/rate leaky bucket parameter.
  - A Talker who sends 2 600-byte frames each 120 µs to CQF running at T_C = 180 µs has to reserve 4*600 bytes per 180 µs.
- Typically, a much better idea would be to reserve 3 600-byte frames per 180 µs, and use a small ”ingress conditioner” = the 802.1Qci counter with a small FIFO.

More work for 802.1 to consider: “Ingress conditioning”

- Suppose you have a non-TSN bridge in the middle of a network.
- Suppose you have a Stream that passes from TSN-aware to non-TSN aware and back to TSN-aware.
- Then, **if and only if**:
  - The latency of the Stream through the non-TSN-aware device has an upper bound on its latency (e.g., the upper bound computed for 802.1CM); and
  - The TSN-aware re-entry device has an “ingress conditioner” with a FIFO sufficient to accommodate the non-TSN device’s latency variation; then
  - The TSN Stream can get a bounded latency.
- Presumably, one uses a central controller to make the reservations.

More work for 802.1 to consider: extra buffering and shaping

- P802.1Qdd Resource Allocation Protocol plans to support reservations that are split/merged for 802.1CB FRER.
- This results in a well-known reservation/buffering problem (C.9 of IEEE Std 802.1CB-2017)
- We have no place (that I know of) in the queuing model of 802.1Q for the buffers required to handle this issue. (It's not difficult – just not done.)
- Not so well-known is how one maintains/calculates/ensures bounded latency when multiple Streams are aggregated into a single Stream, and that aggregation is then demultiplexed.

More work for 802.1 to consider: reordering

- DetNet has looked at the packet discard (as in 802.1CB), noticed that it would be useful, at that point, to offer a service that delivers the packets in order.
- We have noticed extra buffers are needed for the split/merge buffering problem.
- Oddly, the extra buffers are exactly what you need to do re-ordering.
- Should we offer that (presumably, in a .1CB amendment)?

How to proceed

- Contributions? (Duh! That's where it all starts?)
- More 802.1 standards?
- Whitepapers on 802.1 web site?
- Leave it to DetNet?

Thank you