                    UDP Options for Bounded Latency
              draft-qiang-tsvwg-udp-options-bounded-latency-00

Abstract

   This document explores the use of UDP options for packet forwarding
   with bounded latency.

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 September 9, 2019.

Table of Contents

1. Introduction

   More and more applications require for deterministic forwarding as
   summarized in [draft-ietf-detnet-use-cases]. Deterministic
   forwarding refers to the packet forwarding with bounded latency, as
   well as low packet loss. In current DetNet's discussion, low packet
   loss is mainly achieved through PREOF (Packet Replication,
   Elimination, and Ordering Functions)
   [draft-ietf-detnet-architecture]. Supporting PREOF through UDP
   options is out of this document's scope. This document only explores
   the use of UDP options for bounded latency forwarding purposes.

   A lot of bounded latency forwarding schemes have been proposed such
   as Time Aware Shaping [IEEE802.1Qbv], Cyclic Queuing and Forwarding
   [IEEE802.1Qch], Scalable Deterministic Forwarding
   [draft-qiang-detnet-large-scale-detnet], and SR based bounded latency
   [draft-chen-detnet-sr-based-bounded-latency]. These schemes require
   more or less specific information, whether configured through control
   plane or carried in data plane. In MPLS data plane, there is enough
   space in MPLS Label to carry information for bounded latency
   forwarding. However in an IP data plane, there is no enough space in
   the IP common header. To that aim, UDP options
   [draft-ietf-tsvwg-udp-options] are an interesting channel to
   investigate.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
"SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and
"OPTIONAL" in this document are to be interpreted as described in BCP
14 [RFC2119][RFC8174] when, and only when, they appear in all
capitals, as shown here.

1.2. Terminology & Abbreviations

2. Overview of Bounded Latency Forwarding Schemes

In order to figure out what information needs to be carried in an UDP
option, this section briefly analyzes existing bounded latency
forwarding schemes. More details about forwarding schemes themselves
can be found in referenced documents. Generally speaking, in order
to ensure end-to-end bounded latency, each hop along the way should
be latency-bounded. Different forwarding schemes are actually
different packet scheduling methods.

2.1. Time Aware Shaping

In Time Aware Shaping [IEEE802.1Qbv], each port has several
deterministic forwarding queues. Each queue is controlled by a gate.
When a gate opens, packets in the associated queue will be
transmitted. Otherwise, packets should wait in the queue. A
controller calculates the global optimal gate control list for every
port as depicted in Figure 1.

```
      Queue 0    Queue 1         Queue 7

      +------+    +------+        +------+    ^^^^^^^^^^^^^^^^^^^^^
      |------|    |------|        |------|    ^ Gate Control List ^
      |------|    |------|        |------|    ^                   ^
      |------|    |------|        |------|    ^     1000 0000     ^
      |------|    |------|   ....  |------|    ^                   ^
      |------|    |------|        |------|    ^     0001 0000     ^
      |------|    |------|        |------|    ^                   ^
      |------|    |------|        |------|    ^     0000 0001     ^
      |------|    |------|        |------|    ^                   ^
      +--+---+    +--+---+        +--+---+    ^         .         ^
         |           |              |        ^         .         ^
         |           |              |        ^         .         ^
      +--+---+    +--+---+        +--+---+    ^                   ^
      |Gate=1|    |Gate=0|   ....  |Gate=0|    ^                   ^
      +--+---+    +--+---+        +--+---+    ^    1: Gate Open    ^
         |           |              |        ^                   ^
         |           |              |        ^    0: Gate Close   ^
      +--+----------+----------------+---+    ^                   ^
      |           Transmission           |    ^^^^^^^^^^^^^^^^^^^^^
      +----------------------------------+
```
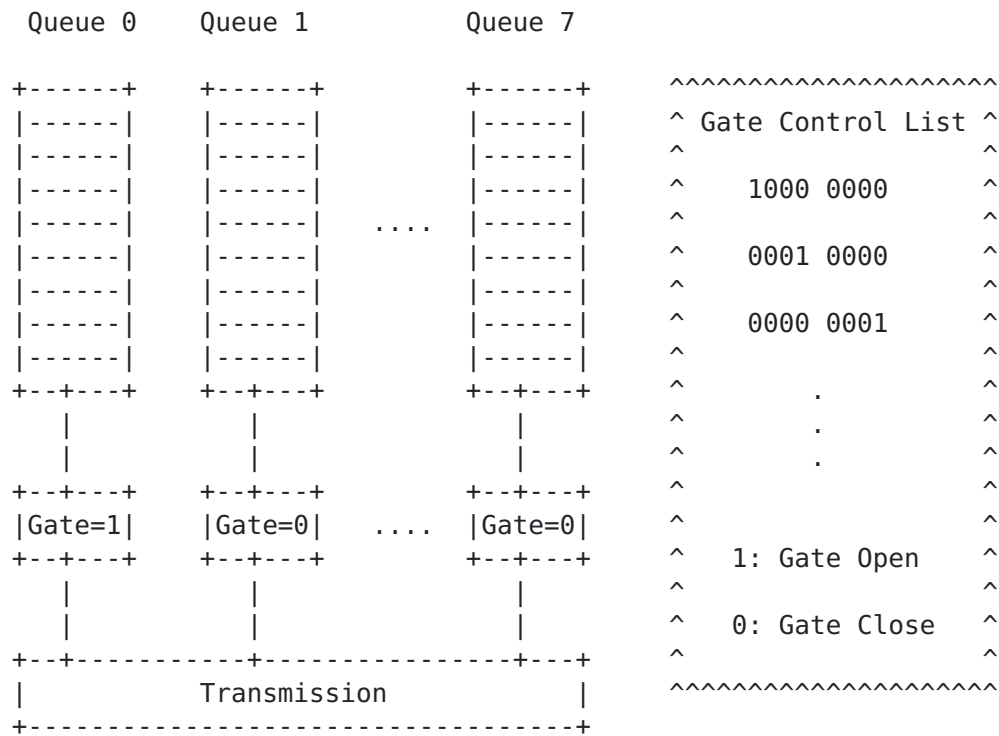
                   Figure 1: Time Aware Shaping

   In Time Aware Shaping, the scheme specific information is "Gate
   Control List". Normally, the gate control list is configured through
   control plane. There is no need for an extension for Time Aware
   Shaping scheme.

2.2. Cyclic Queuing and Forwarding

   In Cyclic Queuing and Forwarding (CQF)[IEEE802.1Qch], all nodes are
   time-synchronized (i.e., time is divided into the same length cycle,
   and all nodes' cycles start at the same timing). Figure 2 explains
   the CQF scheme, packets sent out at a cycle will arrive at downstream
   node at the same cycle, and resent out at the next cycle. As a
   result, once source node sends out packets at a cycle, then the
   receiving cycle of these packets at destination node can be
   determined -- that also means the end-to-end latency is bounded.

```
               Upstream Node |  cycle x  | cycle x+1 |
                             +-----------+-----------+
                              \
                               \packet
                                \receiving
                                 \
               Downstream Node |      V    | cycle x+1 |
                             +-----------+-----------+
                               cycle x       \packet
                                              \sending
                                               \
                                                \
                                                 V
```
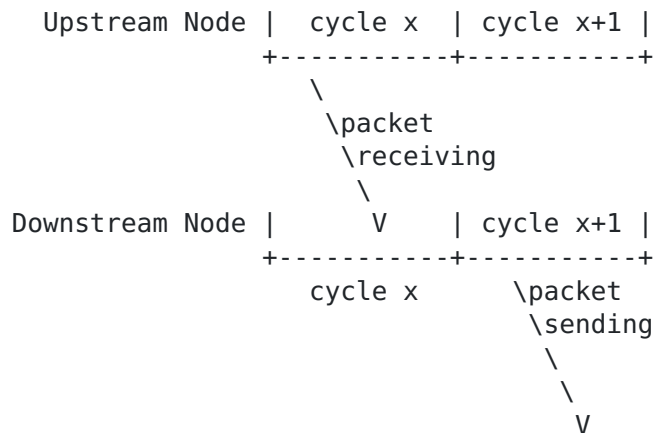
Figure 2: Cyclic Queuing and Forwarding

In CQF, the scheme specific information is "sending cycle" at each
hop. This specific information is actually indicated by the
receiving cycle--each hop can determine the sending cycle of a packet
based on the time this packet be received. Hence no need for an
extension to explicitly carry any scheme specific information in data
plane.

2.3. Scalable Deterministic Forwarding (SDF)

Considering a large scale network deployment, the scalable
deterministic forwarding [draft-qiang-detnet-large-scale-detnet]
scheme makes extensions based on CQF. In scalable deterministic
forwarding (SDF) scheme, all nodes are frequency-synchronized (i.e.,
time is divided into the same length cycle, but different nodes'
cycles may start at different timing). As shown in Figure 3, at a
single cycle y, downstream node may receive packets that were sent
out by upstream node in two different cycles (i.e., cycle x and cycle
x+1). There are two different queues inside the downstream node to
buffer received packets: one is for packets sent out at cycle x,
another one is for packets sent out at cycle x+1. The downstream
node needs to decide which queue a packet should enter based on its
sending cycle at upstream node.

```
            Upstream Node  |  cycle x  | cycle x+1 |
                           +-----------+-----------+
                                \       \
                                 \        \packet
                                  \        \receiving
            Downstream Node       | V      V |           |
                           +-----------+-----------+
                               cycle y    cycle y+1 \ packet
                                                     \ sending
                                                      \
                                                       V
```
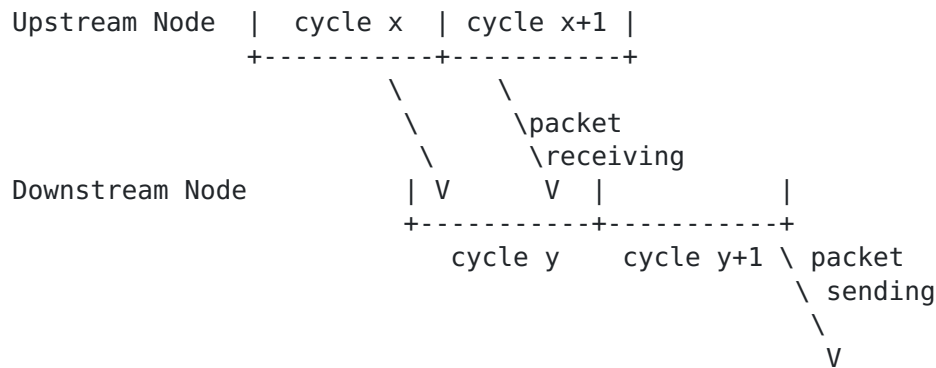
Figure 3: Scalable Deterministic Forwarding

In SDF, the scheme specific information is also "sending cycle".
Since a node may receive packets with different sending cycles within
a single cycle, the sending cycle information should be explicitly
carried with packet. Hence there is a need for an extension to carry
the sending cycle information in IP data plane. The use of UDP
options is a good candidate given the target applications rely upon
an UDP transport.

2.4. SR based Bounded Latency

SR [RFC8402] based bounded latency has similar idea as SDF
(Section 2.3), the main difference is that the "sending cycle"
information is pre-calculated at controller and carried with packet
in MPLS labels. Hence no need for UDP option extension in IP data
plane.

2.5. Other Bounded Latency Forwarding Schemes

TBD

3. UDP Option Extension for Bounded Latency Forwarding

After analyzing the existing bounded latency forwarding schemes, we
found that there are some scheme specific information need to be
carried in data plane. IP common header has very limited space, it
is difficult to find enough bits in IP common header to carry these
scheme specific information. UDP options defined below is a possible
solution for this situation.

```
           Kind       Length         Meaning
           ---------------------------------------------
           127        1              Sending Cycle (SC)
```

Figure 4: UDP Option Extension

3.1. Sending Cycle Option

The Sending Cycle (SC) option includes 1-byte cycle identifier filed
that indicates the cycle a packet be sent out from upstream node as
shown in Figure 5. Normally, the first cyclic forwarding aware node
(e.g., edge gateway) is responsible for creating the SC option, and
setting the Cycle_Identifier filed to be the corresponding cycle that
the packet be sent out. The sending cycle information can help
downstream node to determine when the packet should be resent out.

After receiving packet from upstream node, the downstream node should
interpret the SC option and take an appropriate measures, like
entering a certain queue, or re-sending out immediately, or others.
If there is only one SC option carried in a packet, the
Cycle_Identifier filed may need to be rewritten before re-sending
out.

One way to relieve in-transit rewriting is to pre-compute the sending
cycles along the way, then carry them with multiple SC options.

```
         +----------+---------+-------------------+
         | Kind=127 |  Len=1  |  Cycle_Identifier |
         +----------+---------+-------------------+
```

Figure 5: Sending Cycle Option

3.2. UDP Option for Other Purposes

TBD

[Note: As more bounded latency forwarding schemes are developed, more
UDP option extensions may be required.]

4. IANA Considerations

This document makes no request of IANA.

5. Security Considerations

TBD

6. Acknowledgements

   The Authors would like to thank Mohamed Boucadair, Christian
   Jacquenet and David Black for their review, suggestion and comments
   to this document.

7. Normative References

   [draft-chen-detnet-sr-based-bounded-latency]
              "SR based Bounded Latency",
              <https://datatracker.ietf.org/doc/
              draft-chen-detnet-sr-based-bounded-latency/>.

   [draft-ietf-detnet-architecture]
              Sca, "DetNet Architecture",
              <https://datatracker.ietf.org/doc/
              draft-ietf-detnet-architecture/?include_text=1>.

   [draft-ietf-detnet-use-cases]
              Sca, "DetNet Use Cases",
              <https://datatracker.ietf.org/doc/
              draft-ietf-detnet-use-cases/>.

   [draft-ietf-tsvwg-udp-options]
              "Transport Options for UDPE",
              <https://tools.ietf.org/html/
              draft-ietf-tsvwg-udp-options-05>.

   [draft-qiang-detnet-large-scale-detnet]
              "Large-Scale Deterministic Network",
              <https://datatracker.ietf.org/doc/
              draft-qiang-detnet-large-scale-detnet/>.

   [IEEE802.1Qbv]
              "Enhancements for Scheduled Traffic", 2016.

   [IEEE802.1Qch]
              "Cyclic Queuing and Forwarding", 2016.

   [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>.

   [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC
              2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174,
              May 2017, <https://www.rfc-editor.org/info/rfc8174>.

   [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

   Li Qiang (editor)
   Huawei
   Beijing
   China

   Email: qiangli3@huawei.com

   Toerless Eckert
   Huawei USA - Futurewei Technologies Inc.
   2330 Central Expy
   Santa Clara 95050
   USA

   Email: tte+ietf@cs.fau.de