Network Working Group C. Huitema
Internet-Draft Private Octopus Inc.
Intended status: Experimental January 3, 2020
Expires: July 6, 2020

Quic ACK Timestamps For Measuring One-Way Delays
draft-huitema-quic-1wd-00

Abstract

The QUIC extension for one way delay mesurements adds a single timestamp to the ACK frame. The timestamp is set to the number of microseconds from the beginning of the connection to the time at which the ACK frame is sent. The draft defines the "enable_one_way_delay" transport parameter for negotiating the extension, and two new frame types for timestamped ACK with or without ECN counts.

Status of This Memo

Copyright Notice

Table of Contents

1. Measuring One-Way Delays

The QUIC Transport Protocol [I-D.ietf-quic-transport] provides a secure, multiplexed connection for transmitting reliable streams of application data. The algorithms for QUIC Loss Detection and Congestion Control [I-D.ietf-quic-recovery] use measurement of Round Trip Time (RTT) to determine when packets should be retransmitted. RTT measurements are useful, but there are however many cases in which more precise One-Way Delay (1WD) measurements enable more efficient Loss Detection and Congestion Control.

An example would be the Low Extra Delay Background Transport (LEDBAT) [RFC6817] which uses variations in transmission delay to detect competition for transmission resource. Experience shows that while LEDBAT may be implemented using RTT measurements, it is somewhat inefficient because it will cause unnecessary slowdowns in case of queues or delayed ACKs on the return path. Using 1WD solves these issues. Similar argument can be made for most delay-based algorithms.

We propose to enable one way delay measurements in QUIC by extending the ACK or ACK-ECN format with a timestamp field. The use of this extended format is negotiated with a transport parameter, "enable_one_way_delay". When the extension is negotiated by both parties, the format of the ACK and ACK ECN frames is extended with the timestamp field.

1.1. Terms and Definitions

The keywords "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.

2. Specification

The enable_one_way_delay transport parameter used for negotiating the extension is defined in Section 2.1. The Timestamped ACK frame format is defined in Section 3.

2.1. Negotiation

The one way delay extension is negotiated using a transport parameter:

enable_one_way_delay (TBD): The enable one-way delay transport parameter is included if the endpoint support one way delay measurements for this connection. This parameter has a zero-length value.

Negotiation is successful if both peers support include this parameter in their transport parameter message. If negotiation is successful the peers MUST send Timestamped ACK instead of the standard ACK frames in all 1RTT protected packets. The peers MUST NOT send Timestamped ACK in packets of other types, such as Initial, Handshake or 0RTT protected packet.

Receiving a Timestamped ACK frame when not expected MUST be treated as a Protocol Error. Similarly, receiving a standard ACK frame when a Timestamped ACK frame is expected MUST be treated as a Protocol Error.

3. Timestamped ACK format

Timestamped ACK are identified by the frame type:

o Timestamped_ACK (TBD, TBD+1)

If the frame type is TBD+1, Timestamped ACK frames also contain the sum of QUIC packets with associated ECN marks received on the connection up until this point.

The format of the Timestamped ACK frames is similar to that of the standard ACK Frames defined in section 19.3 of

[I-D.ietf-quic-transport], with the addition of the Time Stamp parameter.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Largest Acknowledged (i)                ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Time Stamp (i)                     ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          ACK Delay (i)                      ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       ACK Range Count (i)                   ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       First ACK Range (i)                   ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          ACK Ranges (*)                     ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          [ECN Counts]                       ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: ACK Frame Format with Time Stamp

The timestamp encodes the number of microseconds since the beginning of the connection, as measured by the peer at the time at which the ACK is sent. It is encoded using the exponent selected by the peer in the ack_delay_exponent. The exponent reduced time stamp is encoded in the ACK frame as a variable length integer.

3.1. RTT Measurements

RTT measurements are performed as specified in Section 4 of [I-D.ietf-quic-recovery], without reference to the Timestamp parameter of the Timestamped ACK frames.

3.2. One-Way Delay Measurements

An endpoint generates a One Way Delay Sample on receiving a TimeStamped ACK frame that meets the following two conditions:

- the largest acknowledged packet number is newly acknowledged, and

- at least one of the newly acknowledged packets was ack-eliciting.

The One Way Delay sample, latest_1wd, is generated as the time elapsed since the largest acknowledged packet was sent, corrected for the difference between local time at the sending peer and connection time at the receiving peer, phase_shift.

```
latest_1wd = time_stamp - send_time_of_largest_acked - phase_shift
```

By convention, the phase_shift is estimated upon reception of the first RTT sample, first_rtt. It is set to:

```
phase_shift = time_stamp - send_time_of_largest_acked - latest_rtt/2
```

In that formula, we assume that the local time are measured in microseconds since the beginning of the connection.

We understand that clocks may drift over time, and that simply estimating a phase shift at the beginning of a connection may be too simplistic for long duration connections. Implementations MAY adopt different strategies to reestimate the phase shift at appropriate intervals. Specifying these strategies is beyond the scope of this document.

4. Discussion

This document makes a series of choices in implementing one way delay measurements. The two important choices are the decision to only have one timestamp per Timestamped ACK frame, and to use new frame types for the Timestamped ACK frames.

In theory, it would be possible to augment the ACK frame format and document the delay of every frame received. The current proposal is simpler, and creates less overhead. It is also sufficient for the intended usage, which is to improve the efficiency of congestion control protocols.

Once the extension is negotiated, it would be possible in theory to reuse the existing ACK frame type and just associate it with the timestamp format. This would spare us the need to reserve two frame type numbers for the new format. However, this simplification would introduce a dependency between the state of the parser and the state of the connection. This dependency is hard to manage in off-line usages such as parsing log files.

5. Security Considerations

The Timestamp value in the Timestamp ACK is asserted by the sender of the ACK. Adversarial peers could chose values of the time stamp designed to exercise side effects in congestion control algorithms or other algorithms relying on the one-way delays. This can be mitigated by running plausibility checks on the received values. For example, each peer can maintain statistics not just on the One Way Delays, but also on the differences between One Way Delays and RTT, and detect outlier values. Peers can also compare the differences

between timestamps of successive acknowledgements and the differences between the sending times of corresponding packets, and detect anomalies if the delays between acknowledging packets appears shorter than the delays when sending them.

6. IANA Considerations

This document registers a new value in the QUIC Transport Parameter Registry:

Value: TBD (using value 0x10DE in early deployments)

Parameter Name: enable_one_way_delay

Specification: Indicates that the connection should use TimeStamped ACK frames

This document also registers a new value in the QUIC Frame Type registry:

Value: 0x34 and 0x35 (if this document is approved)

Frame Name: Timestamped ACK

Specification: ACK frames augmented with a timestamp

7. References

7.1. Normative References

[I-D.ietf-quic-recovery]
Iyengar, J. and I. Swett, "QUIC Loss Detection and Congestion Control", draft-ietf-quic-recovery-24 (work in progress), November 2019.

[I-D.ietf-quic-transport]
Iyengar, J. and M. Thomson, "QUIC: A UDP-Based Multiplexed and Secure Transport", draft-ietf-quic-transport-24 (work in progress), November 2019.

[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>.

7.2. Informative References

[RFC6817] Shalunov, S., Hazel, G., Iyengar, J., and M. Kuehlewind, "Low Extra Delay Background Transport (LEDBAT)", RFC 6817, DOI 10.17487/RFC6817, December 2012, <https://www.rfc-editor.org/info/rfc6817>.

Author's Address

Christian Huitema
Private Octopus Inc.
427 Golfcourse Rd
Friday Harbor WA 98250
U.S.A

Email: huitema@huitema.net