Independent Submission
Request for Comments: 8774
Category: Informational
ISSN: 2070-1721

M. Welzl
University of Oslo

1 April 2020

The Quantum Bug

Abstract

The age of quantum networking is upon us, and with it comes "entanglement": a procedure in which a state (i.e., a bit) can be transferred instantly, with no measurable delay between peers. This will lead to a perceived round-trip time of zero seconds on some Internet paths, a capability which was not predicted and so not included as a possibility in many protocol specifications. Worse than the millennium bug, this unexpected value is bound to cause serious Internet failures unless the specifications are fixed in time.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This is a contribution to the RFC Series, independently of any other RFC stream. The RFC Editor has chosen to publish this document at its discretion and makes no statement about its value for implementation or deployment. Documents approved for publication by the RFC Editor are not candidates for any level of Internet Standard; see Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc8774.

Table of Contents

1. Introduction

[RFC6921] discusses faster-than-light communication, where packets arrive before they are sent. While it is amusing to entertain the possibility of time travel, we have to accept the cold facts: time travel will never work (or it would already have been used). Quantum networking, however, is an entirely different matter -- commercial products are already available, and quantum networks will without a doubt become the prevalent Internet link-layer technology across the globe within the next five to ten years.

With the help of entanglement, implemented in quantum repeaters, quantum networks can transfer information faster than ever before: a state can be transmitted over a long distance instantly, with no delay. This is so cool that it is also called (and, by some, mistaken for) teleportation. If a path between a sender and a receiver is fully quantum-ized, the measured one-way delay (OWD) will be zero. What's more, assuming that there are blazing fast quantum computers involved on both ends, the processing time will be well below anything measurable; hence, even the round-trip time (RTT) will be zero in these scenarios.

In today's Internet, only very few protocols are prepared for such "0-RTT" situations (e.g., TCP with "TCP Fast Open" (TFO) [RFC7413], TLS 1.3 [RFC8446], and QUIC [QUIC-TRANS]). Many others will fail in interesting ways; we coin the term "Quantum Bug" for such failures. In the following section, we will discuss some examples of Quantum Bugs.

2. Protocols and Protocol Mechanisms That Will Fail

The number of protocols and protocol mechanisms that will fail in the face of a zero RTT is too large to report here; we are truly heading towards something close to an Internet meltdown. We can only provide some guidance to those who hunt for the Quantum Bug, by discussing examples of specification mistakes that will need to be fixed.

2.1. LEDBAT

The Low Extra Delay Background Transfer (LEDBAT) congestion control mechanism [RFC6817] is a very interesting failure case: designed to "get out of the way" of other traffic; it will end up sending as fast as possible. Specifically, when the algorithm described in Section 2.4.2 of [RFC6817] obtains a delay sample, it updates a list of base delays that will all become 0 and current delays that will also all become 0. It calculates a queuing delay as the difference between the current delay and the base delay (resulting in 0) and keeps increasing the Congestion Window (cwnd) until the queuing delay

reaches a predefined parameter value TARGET (100 milliseconds or less).

A TARGET value of 100 milliseconds will never be reached, because the queuing delay does not grow when the sender increases its cwnd; this means that LEDBAT would endlessly increase its cwnd, limited only by the number of bits that are used to represent cwnd. However, given that TARGET=0 is also allowed, this parameter choice may seem to be a way out. Always staying at the target means that the sender would maintain its initial cwnd, which should be set to 2. This may seem like a small number, but remember that cwnd is the number of bytes that can be transmitted per RTT (which is 0). Thus, irrespective of the TARGET value, the sender will send data as fast as it can.

2.2. Multipath TCP (MPTCP)

The coupled congestion control mechanism proposed for MPTCP in [RFC6356] requires calculating a value called "alpha". Equation 2 in [RFC6356] contains a term where a value called "cwnd_i" is divided by the square of the RTT, and another term where this value is divided by the RTT. Enough said.

2.3. RTP Circuit Breakers

The RTP Circuit Breakers [RFC8083] require calculation of a well-known equation which yields the throughput of a TCP connection:

$$X = \frac{s}{Tr*\sqrt{2*b*p/3}+(t\_RTO * (3*\sqrt{3*b*p/8}) * p * (1+32*p*p))}$$

where Tr is the RTT and t_RTO is the retransmission timeout of TCP (we don't need to care about the other variables). As we will discuss in Section 3, t_RTO is lower-bounded with 1 second; therefore, it saves us from a division by zero. However, there is also a simplified version of this equation:

$$X = \frac{s}{Tr*\sqrt{2*b*p/3}}$$

Unfortunately, [RFC8083] states: "It is RECOMMENDED that this simplified throughput equation be used since the reduction in accuracy is small, and it is much simpler to calculate than the full equation." Due to this simplification, many multimedia applications will crash.

3. What can be done?

Fear not: when everything else fails, TCP will still work. Its retransmission timeout is lower-bounded by 1 second [RFC6298]. Moreover, while its cwnd may grow up to the maximum storable number, data transmission is limited by the Receiver Window (rwnd). This means that flow control will save TCP from failing.

From this, we can learn two simple rules: lower-bound any values

calculated from the RTT (and, obviously, do not divide by the RTT), and use flow control. Specifications will need to be updated by fixing all RTT-based calculations and introducing flow control everywhere. For example, UDP will have to be extended with a receiver window, e.g., as a UDP option [UDP-OPT].

4. Conclusion

We are in trouble, and there is only one way out: develop a comprehensive list of all RFCs containing "0-RTT" mistakes (taking [RFC2626] as a guideline), and update all code. This needs to happen fast, the clock is ticking. Luckily, if we are too slow, we will still be able to use TCP to access the specifications. With DNS over TCP [RFC7766], name resolution to find the server containing the specifications should also work.

5. IANA Considerations

This document has no IANA actions.

6. Security Considerations

Flow control must be used on 0-RTT paths, or else an attacker can completely overwhelm a sender with data in a denial-of-service (DoS) attack within an instant. Flow control will need to be added to protocols that do not currently have it, such as UDP or ICMP. IPv6 will not save us.

7. References

7.1. Normative References

[RFC2626] Nesser II, P., "The Internet and the Millennium Problem (Year 2000)", RFC 2626, DOI 10.17487/RFC2626, June 1999, <https://www.rfc-editor.org/info/rfc2626>.

[RFC6921] Hinden, R., "Design Considerations for Faster-Than-Light (FTL) Communication", RFC 6921, DOI 10.17487/RFC6921, April 2013, <https://www.rfc-editor.org/info/rfc6921>.

7.2. Informative References

[QUIC-TRANS] Iyengar, J. and M. Thomson, "QUIC: A UDP-Based Multiplexed and Secure Transport", Work in Progress, Internet-Draft, draft-ietf-quic-transport-27, 21 February 2020, <https://tools.ietf.org/html/draft-ietf-quic-transport-27>.

[RFC6298] Paxson, V., Allman, M., Chu, J., and M. Sargent, "Computing TCP's Retransmission Timer", RFC 6298, DOI 10.17487/RFC6298, June 2011, <https://www.rfc-editor.org/info/rfc6298>.

[RFC6356] Raiciu, C., Handley, M., and D. Wischik, "Coupled Congestion Control for Multipath Transport Protocols",

RFC 6356, DOI 10.17487/RFC6356, October 2011, <https://www.rfc-editor.org/info/rfc6356>.

[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>.

[RFC7413] Cheng, Y., Chu, J., Radhakrishnan, S., and A. Jain, "TCP Fast Open", RFC 7413, DOI 10.17487/RFC7413, December 2014, <https://www.rfc-editor.org/info/rfc7413>.

[RFC7766] Dickinson, J., Dickinson, S., Bellis, R., Mankin, A., and D. Wessels, "DNS Transport over TCP - Implementation Requirements", RFC 7766, DOI 10.17487/RFC7766, March 2016, <https://www.rfc-editor.org/info/rfc7766>.

[RFC8083] Perkins, C. and V. Singh, "Multimedia Congestion Control: Circuit Breakers for Unicast RTP Sessions", RFC 8083, DOI 10.17487/RFC8083, March 2017, <https://www.rfc-editor.org/info/rfc8083>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

[UDP-OPT] Touch, J., "Transport Options for UDP", Work in Progress, Internet-Draft, draft-ietf-tsvwg-udp-options-08, 12 September 2019, <https://tools.ietf.org/html/draft-ietf-tsvwg-udp-options-08>.

Author's Address

Michael Welzl
University of Oslo
PO Box 1080 Blindern
N-0316 Oslo
Norway

Phone: +47 22 85 24 20
Email: michawe@ifi.uio.no