



## QUIC RTT estimation without Latency Spin Bit

### Abstract

**QUIC** is a secure general-purpose, encrypted, multiplexed, and low-latency transport protocol designed from the ground up to improve transport performance for HTTPS traffic. QUIC has recently (May 2021) become RFC standard (RFC 9000) and is expected to become the dominant transport protocol in the Internet over TCP. Most of QUIC packets are encrypted. Not only the payload is encrypted but also most of the header. This situation makes it very difficult for the ISP (Internet Service Provider) to monitor the network and enforce regulatory measures to keep the network up and running. One important parameter for ISPs to monitor network health is RTT estimation. When RTT rise too high for many flows it may indicate that queues build-up and an up-coming network collapse. In such situations the ISP may limit heavy users and keep the network alive. Therefore, an efficient estimation of RTT is an important network management. As QUIC is almost fully encrypted, there are only few unencrypted bits, one of them is the Latency SPIN Bit which was added to the protocol especially for RTT estimation.

Latency SPIN Bit operation:

- When a server sends a packet, it sets the spin bit to the spin bit on the last packet it received from the client.
- When a client sends a packet, it sets the spin bit to the inverse of the spin bit on the last packet it received from the server.

Therefore, the SPIN bit has an **edge** (transition from “1” to “0” or from “0” to “1”) every RTT and an observer can estimate the RTT by measuring the time between edges of a connection.

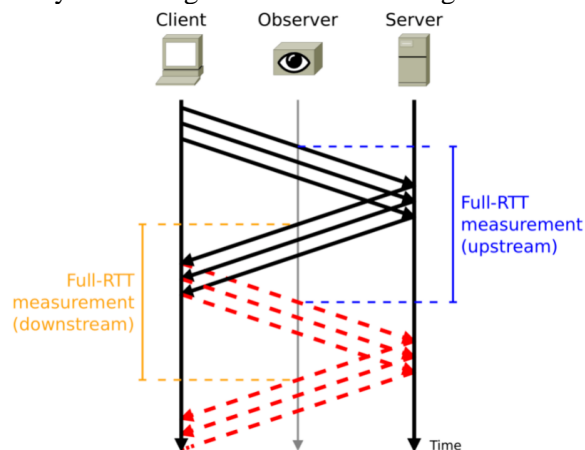


Figure 1: Latency SPIN bit operation

### Problem

It is hard for a middle box observer to estimate the Round Trip Time of of a QUIC connection.



## Project objective

Set a middle box observer that estimates the RTT of connections using a Latency SPIN bit.

## Project overview

1. Ramp-up previous project (that measured RTT by observing the Latency SPIN Bit) as a starting point.
2. Suggest at least two algorithms to estimate the RTT without the Latency Spin Bit.
3. Implement two of the algorithms.
4. Define KPIs and compare the two implementations to the client and servers' internal RTT estimation.

## Notes

- The above list is an estimate. Goals and tasks might be modified during the first few weeks of the projects before the finalization of High Level Design Document.

General requirements for all LCCN Projects are specified at the lab website:

<https://lcn.cs.technion.ac.il/lab-courses/>

## Prerequisites:

1. Introduction to computer networks (236334)
2. Internet Networking (236341)

Instructor: Eran Tavor (tavran@cs.technion.ac.il)