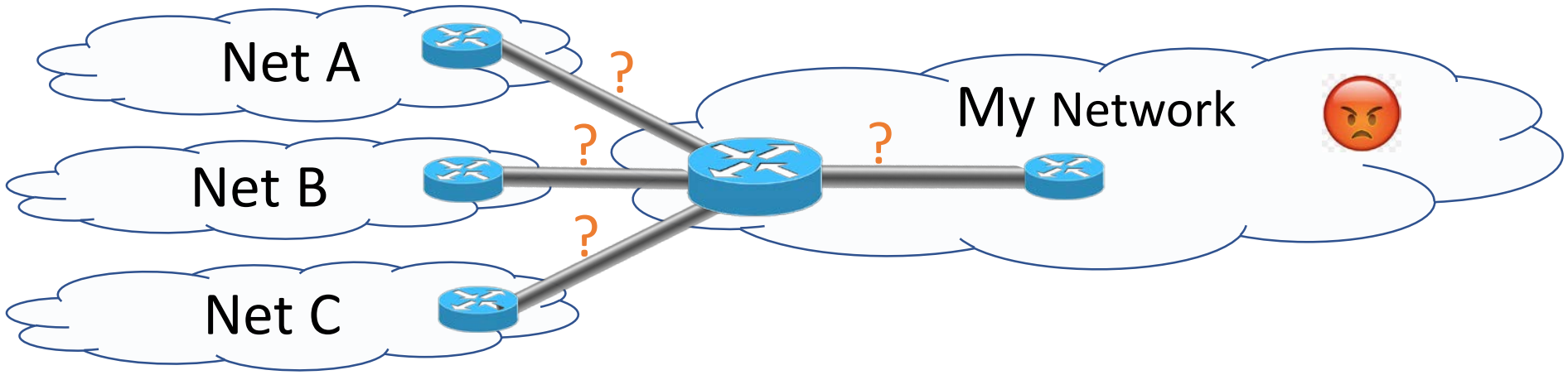
Loss Bits Extension

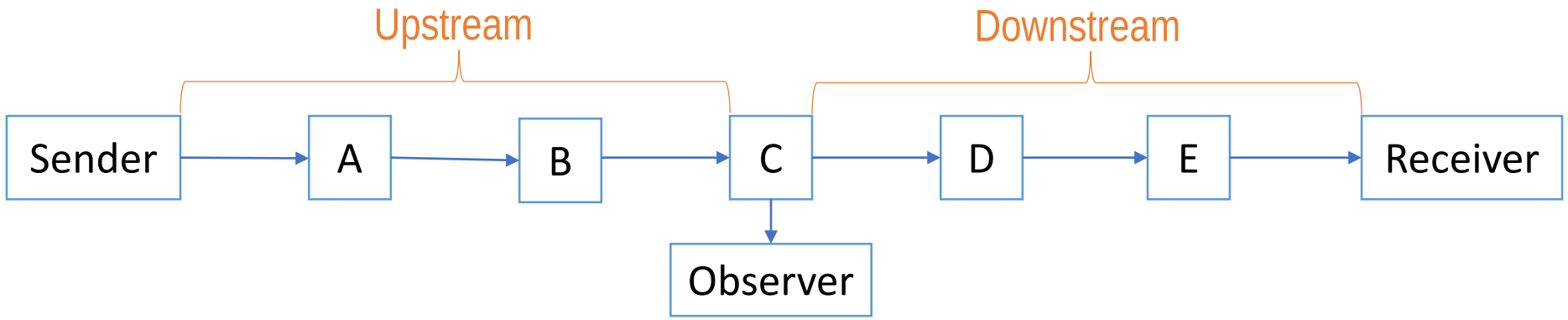
draft-ferrieuxhamchaoui-quic-lossbits

February 5-6, 2020
QUIC Interim, Zurich

The Problem – Find source of delay and loss?



Operators must monitor Delay and Loss and address problems quickly



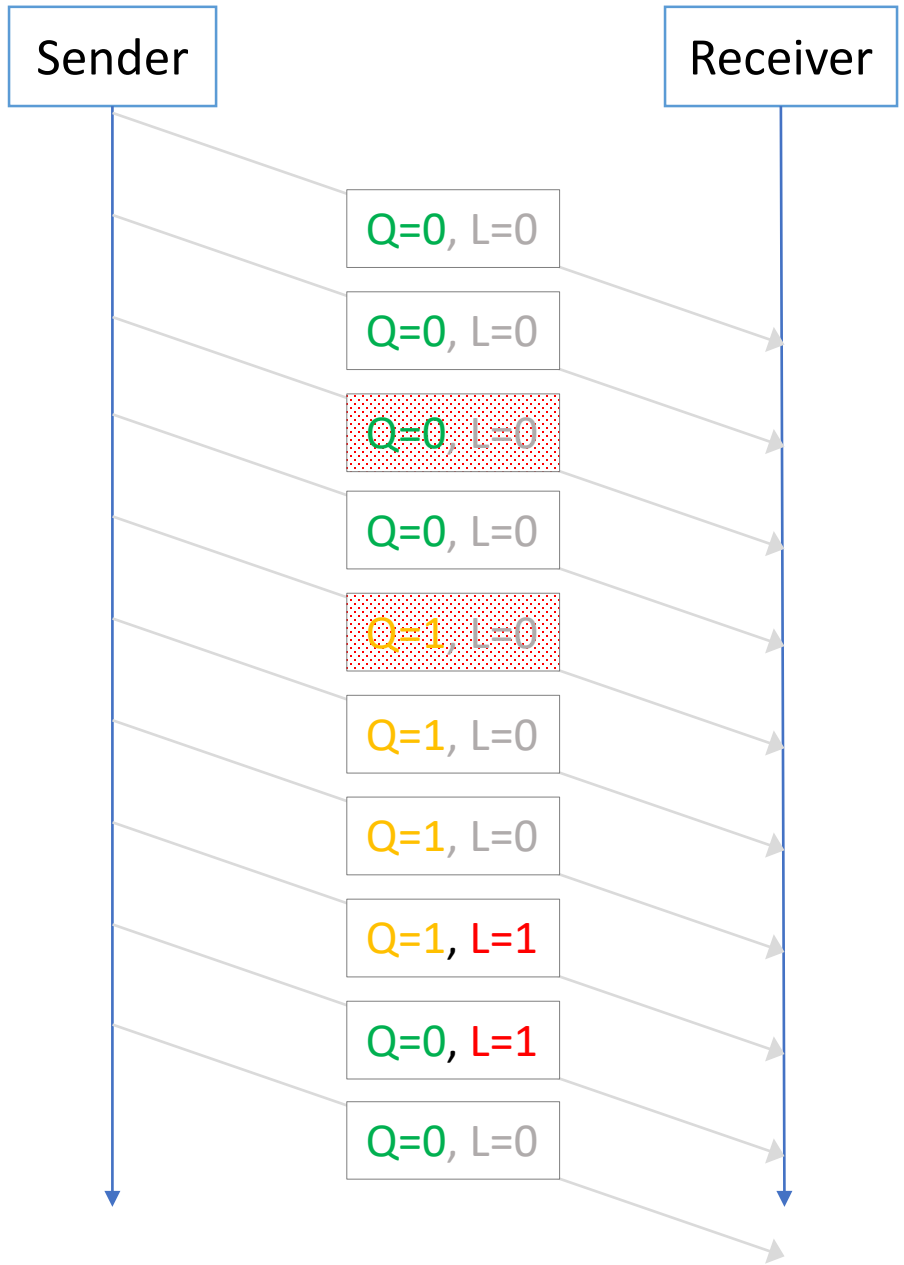
Summary of the Extension

Negotiate:

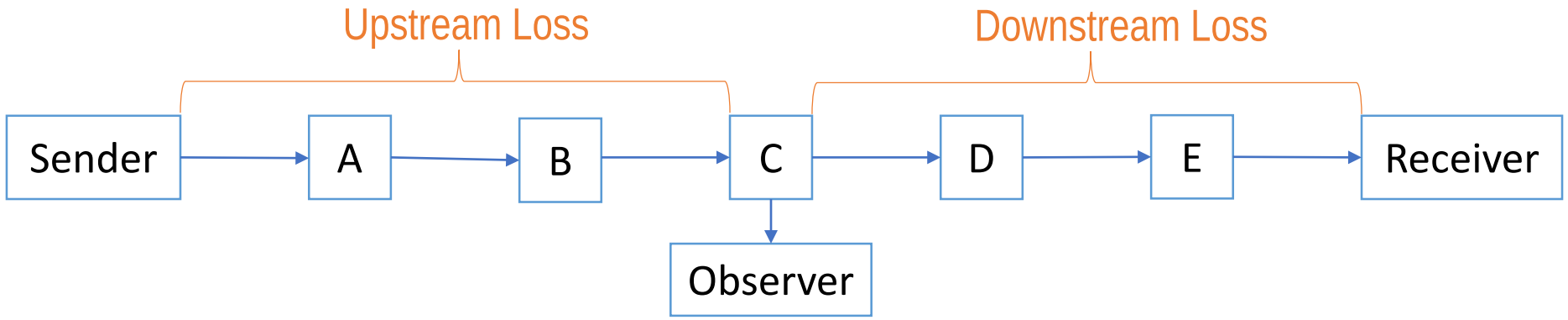
- Short header: **0 1 S R R K P P** → **0 1 S Q L K P P**
- header protection mask: 0x1F → 0x07

- **Q**: The “sQuare signal” bit is toggled every N outgoing packets

- **L**: The “Loss event” bit is 1 when “Unreported Loss Counter” (ULC) > 0
  - ULC is incremented for each packet deemed lost
  - ULC is decremented for each packet sent with L=1



Loss Calculation



- End-to-End loss (e)

$$e = \text{fraction of packets with L=1}$$

- Upstream loss (u)

$$u = 1 - \frac{\text{average \# of observed packets in a block (same Q)}}{\text{size of the block}}$$

- Downstream loss (d)

$$(1-u)(1-d) = 1-e \qquad d = \frac{e-u}{1-u} \approx e-u$$

Negotiation

Goals:

1. Both endpoints must agree
2. Allow peer to send loss bits w/o implementing loss bits yourself

- Transport Parameter: 0x1057 (“LOST”)
  - Value 0: “Peer can send loss bits in short header, but I *will not* do so myself.”
  - Value 1: “Peer can send loss bits in short header, and I *want* to do so myself.”
- No TP from both endpoints → no loss bits in any direction

Privacy, Ossification, Security

Goal: Do not introduce new privacy, security, ossification issues

Privacy

- MUST keep separate loss counters per CID (no cross-CID correlations)

Ossification

- MUST NOT use Loss Bits TP on at least 1/16 of the connections

Security

- Optimistic ACK Attack is easier: attacker cannot simulate a lower rtt, but it might detect and ACK true losses. A sender should shortens Q run length when skipping a packet number (especially if it implements DATAGRAM and a loss-sensitive congestion controller).

More Privacy Risks – Peeling the Onion?

Setup

- Suspect is connecting to an illegal server via Tor
- Attacker is watching traffic to an illegal server
- Attacker is able to induce loss at sender (EM, network level)

Attack

- Attacker induces loss & uses loss signal to confirm a flow from sender

Analysis

- Sender using Tor is likely to disable Loss Bits
- Loss response can also be observed by packet timing w/o loss signal
- Same attack with s/loss/delay/g

Last Slide

Current State

- Prev version deployed for ~1 year on Akamai production servers. Interop w/ Orange on-path observers (presented at IETF-105,106).
- Latest version: have an Interop w/ picoquic and lsquic

Next Steps

- There was interest in Singapore from the community to look at Privacy/Security on the road to adoption.

  Looking for feedback/suggestions/collaboration on Privacy/Security!

  https://github.com/igorlord/draft-ferrieuxhamchaoui-lossbits