

# Hackathon update: Testing QUIC clients with CR-enabled QUIC

# Interop setup

Docker setup with bridge either side of simulation container



- **Path simulator is NS3**

- 600ms delay; 10Mbps
- .. only 5Mbps achieved due to NS3 limitations
- Server is our modified Quiche, which includes CR
  - Previous RTT 600ms; and Jump of 400 packets

# Cloudflare Quiche with CR

These are "work-in-progress" results

	quiche	aiquic	kwic	mvfst	ngtcp	picoquic	quic-go	chrome	lsquic	neqo	msquic
Jump happens?	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Sends Unvalidated Packets	773	191	212	200	191	201	270	95	181	210	138
Validates CWND	SR	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Final CWND	320,323	310,740	258,213	255,655	438,278	469,988	503,106	154,197	411,908	492,626	225,801

Without CR cwnd ~ 80 KB

# Conclusions and Plans

## ***This time we found:***

All clients returned ACKs that left the server perform CR!

Not all clients fully utilised the jump cwnd (aioquic, kwik, mvfst, msquic), but all benefited.

We will follow-up on finding more about these differences.

Conclusion: All clients worked, we can discover more.

## ***Next Hackathon:***

We will replace the "ns3" bottleneck with a "dedicated emulator"

We can include multiple server implementations: PicoQUIC (2 flavours); Cloudflare; Others (do talk to us)

Would like some more interesting work patterns and better performance metrics.

***We'll be back!***