# Making QUIC Deployable

## QUIC, IETF 103, November 2018

Ian Swett

# First Step: IETF Invariants

In London, IETF 101, Invariants were adopted

Provided a guarantee part of QUIC was stable

Chrome implements the Invariants in QUIC v44*

    Public header, including asymmetric CIDs, VN

    Stateless Reset replaces Public Reset

*True varlen CIDs are not yet supported

QUIC

# Deployment Status: In Chrome

Canary: Enabled for all QUIC users

Dev: Enabled for all QUIC users

Beta: Experiment running for >1 month

Stable: Experiment running for ~1 week

# The Good :)

Contacted the middlebox vendor whose QUIC identification caused issues before

They've developed a fix for v44 and deployed it!

No reports of users visible issues

# The Bad :(

Rate of handshake timeouts has increased

Number of Stateless Resets << gQUIC Public Resets

Small, but significant regressions in user metrics


Theory: Some failed handshakes aren't being closed

# What's next?

Fix handshake timeouts ASAP

If draft-17 becomes a deployment draft, Ship It!

QUIC

# Takeaways

Supporting IETF QUIC and gQUIC simultaneously is complex

If you don't have to do it, don't...

Detailed client and server metrics are critical for debugging
hard to repro issues

No obvious signs of middlebox issues with IETF invariants

QUIC