SCONE Hackathon

IETF 123 Madrid

Background

IETF SCONE WG:

- **SCONE** = **S**tandard **CO**mmunication with **N**etwork **E**lements
- Standardizing QUIC-based capability to help adaptive bitrate (video) flows to be aware of network throttling rates.
- https://datatracker.ietf.org/group/scone/about/

Protocol: https://datatracker.ietf.org/doc/draft-ietf-scone-protocol/

SCONE requires compatible end-host (client + server) and in-network implementations.

Code Worked On and Tested

End-host client/server implementations:

- Python aioquic
- Ericsson prototype client/server
- Cloudflare quiche
- Meta prototype in mvfst
- Nokia prototype client/server
- YouTube prototype

In-network implementations:

- Ericsson prototype
- Nokia prototype
- Open-source prototype using eBPF/XDP
  - https://github.com/wesley-eddy/scone-tools

Scripts to orchestrate tests.

(side quest: some work on QUIC spin bit in quiche.)

Summary

~8 in-person individuals participated and several remote, from 6+ companies+universities, including (alphabetically):

- Ericsson
- Meta
- Nokia
- Tiktok
- UCSB
- YouTube/Google

Successful transport parameter negotiation (mvfst & quiche clients w/ Ericsson server).

Successful network element updating of SCONE packets (Ericsson element).

2 github issues posted on the protocol spec I-D for clarification.

Lessons Learned

Opened new github issues on protocol draft:

- Zero-length value for SCONE transport parameter
- Suggestion to add text on UDP checksum update.

Psuedo-SCONEs