IETF 118 Deepspace side meeting

QUIC over TAPS API for Deep Space
TAPSCE

Hackathon Plan

- <What problem were you working on?>
  - Tune QUIC with TAPS API for using QUIC in deep space
    - <Specific problems to solve>
      - Support extremely long delay, no ack, disruption…
- <How you planned to solve it?>
  - implement TAPS on the top of QUIC
  - Identify QUIC and DTN parameters, profiles, features missing to be added

What got done

- <Lessons learned from this hackathon>
  - TAPS and QUIC implementations over python asyncio might be assembled
    - https://github.com/aiortc/aioquic
    - https://github.com/fg-inet/python-asyncio-taps/tree
  - careful-resume and bdp frame might be implemented soon in QUIC
  - We started to merge python-asyncio-taps and aioquic

What is planed

- interop with Christian Quic to Mars implementation
- interop with QUIC implementations suppoting Careful Resume
- <New feedback to take to WG?> <New work to take to WG?>
  - TAPS and QUIC WG : supports whenever extensions to TAPS and QUIC are needed
  - Need feedback from DTN Bundle Protocol Implementations to map parts of DTN using TAPS specifications

Wrap Up

Team members:

Emile Stephan (emile.stephan@orange.com)

Marc Blanchet (marc.blanchet@viagenie.ca)

Max Max Franke (mfranke@inet.tu-berlin.de)

Special thanks to Gorry and Ana from Careful Resume Table, and the visitors from TAPS

Careful Resume

- Key mechanisms at start-up
  - Slow-start limits capacity use to prevent overshooting but impacts latency and other flows.
  - Hystart++ prevents overshooting the bottleneck and congestion, preserving shared capacity.
  - CR speeds up transfers with high data volume over large path capacity compared to large IW.

- Main Features
  - Reuses past parameters for faster connection restart.
  - Sender sets requirements for capacity utilisation.
  - Ensures safe response when capacity/RTT changes.

* From TSVWG WG, IETF-118, Prague, Nov 2023 slides