Priority-aware Forward Error Correction for HTTP

Nooshin Eghbal and Paul Lu
University of Alberta, Canada
eghbal@ualberta.ca

Applied Networking Research Workshop (ANRW), July 2022
Motivation:

● Takeaway Message: To reduce page-load times, use **selective** Forward Error Correction (FEC) **only** for high-priority resources (e.g., HTML, CSS, JS) in HTTP/3.

● Previously:
  ○ HTTP had “complicated” dependency tree prioritization mechanism. Dropped.
  ○ Early QUIC had FEC for *all* data. Too expensive. Gone.

● Now:
  ○ HTTP/3 has simpler Extensible Prioritization Scheme
  ○ HTTP/3 *could* have **selective** FEC to reduce overheads
Implementation and Evaluation

- **Paper:**
  - Evaluation over UDP-based Data Transfer (UDT) protocol using 2D FEC

- **This talk:**
  - Evaluation over ngtcp2/nghttp3 using OpenFEC library
    - Precompute repair data and send as an HTTP resource
    - Decode QUIC frames to recover lost ones
    - OpenFEC supports Reed-Solomon/LDPC FEC

- **Two nodes in Emulab testbed**
  - Using Netem-tc Linux tool to add RTT and loss rate

Priority-aware FEC for HTTP
Emulation Results:

- RTT: 100ms, packet loss: 10%
- Arrival **order** of QUIC frames at HTTP client w/o FEC

![Diagram showing Emulation Results]

Priority-aware FEC for HTTP
Summary and Future Work

- Resource prioritization at HTTP server helps
  - Reduce FEC overhead by considering **only** high priority resources
  - Reduce the page load time by downloading essential resources sooner

- Improving our implementation over QUIC (ngtcp2)
- Studying the congestion control effects of FEC
Thanks!

Questions/suggestions?

eghbal@ualberta.ca

Priority-aware FEC for HTTP