FEC FRAME for WebRTC

Giri Mandyam
Qualcomm Innovation Center
draft-mandyam-rtcweb-fecframe-00
Co authors: M. Luby, T. Stockhammer, C. Foisy
Introduction

• IETF FEC (Forward Error Correction) Framework encapsulated the application of FEC to streaming protocols
  – RFC 6363 describes framework
  – RFC 6364 provides SDP semantics
• FEC FRAME is readily applicable to WebRTC
FEC streaming

- There are multiple standardized FEC codes for streaming
  - Reed-Solomon, Raptor, RaptorQ, LDPC
- FEC is used to protect against packet loss
  - Partition source stream into source blocks of data
    - Partitioning can be done on the fly as the stream becomes available
  - Encoding block = source block + FEC repair
    - FEC repair generated from the source block to provide protection against packet loss
    - Send encoding block for a source block
    - Based on redundancy in sent encoding block, receiver may be able to recover source block when there is packet loss
FEC streaming trade-offs

- Smaller source blocks $\rightarrow$ Better end-to-end latency
- Larger source blocks $\rightarrow$ Better recovery performance
- Less FEC repair $\rightarrow$ Less bandwidth
- More FEC repair $\rightarrow$ Better recovery performance
FEC streaming example

- 2 Mbps H.264 streaming session
- RaptorQ code (RFC 6682) or Reed-Solomon (RFC 6865)
- Target failure to recover source block = $10^{-6}$

<table>
<thead>
<tr>
<th>Packet Loss</th>
<th>Source block stream duration*</th>
<th>Encoding stream rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 %</td>
<td>46.08 ms</td>
<td>2.75 Mbps</td>
</tr>
<tr>
<td>5 %</td>
<td>46.08 ms</td>
<td>3.50 Mbps</td>
</tr>
<tr>
<td>10 %</td>
<td>46.08 ms</td>
<td>4.00 Mbps</td>
</tr>
<tr>
<td>1 %</td>
<td>97.92 ms</td>
<td>2.47 Mbps</td>
</tr>
<tr>
<td>5 %</td>
<td>97.92 ms</td>
<td>2.94 Mbps</td>
</tr>
<tr>
<td>10 %</td>
<td>97.92 ms</td>
<td>3.29 Mbps</td>
</tr>
</tbody>
</table>

* = source block size/source streaming rate
this is a lower bound on, and indicative of, end-to-end latency
What is Requested

• FEC be allowed for WebRTC sessions when both endpoints support
  – Subject to negotiation between endpoints
  – All SDP semantics for WebRTC be compatible with FEC negotiation

• draft-mandyam-rtcweb-fecframe-00 become an RTCWEB Working Group draft