FEC FRAME for Multiple SSRC's

Giri Mandyam
Qualcomm

draft-mandyam-rtcweb-fecframebundledssrc-00

Co authors: M. Luby and T. Stockhammer

Introduction

- IETF FEC (Forward Error Correction) Framework encapsulated the application of FEC to streaming protocols
  - RFC 6363 describes framework
  - RFC 6364 provides SDP semantics
- RTCWeb FEC specification makes use of FEC FRAME
  - Current draft draft-ietf-rtcweb-fec-01
  - “To offer support for a separate FEC stream, the offerer MUST offer one of the formats described in [I-D.ietf-payload-flexible-fec-scheme], Section 5.1, as well as a ssrc-group with "FEC-FR" semantics as described in [RFC5956], Section 4.3.”

Introduction (cont.)

- draft-ietf-rtcweb-fec-01 also states
  - “Support for protecting multiple primary streams with a single FEC stream is complicated by WebRTC's 1-m-line-per-stream policy and requires further study.”
- The intention behind this I.-D. is to commence the necessary further study
  - Provide necessary extensions to FEC FRAME to allow for “bundled” FEC protection
    - An FEC stream that protects multiple RTP streams

Why Examine in Context of WebRTC

- FEC FRAME WG is no longer active
- Specialized topic – may not be worth the overhead of opening (re-opening) a dedicated working group
- WebRTC is an ideal technology to examine and verify bundled FEC protection
  - Multi-stream transmission over RTP

Background - 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 → Better end-to-end latency
- Larger source blocks → Better recovery performance
- Less FEC repair → Less bandwidth
- More FEC repair → Better recovery performance

Technical Description

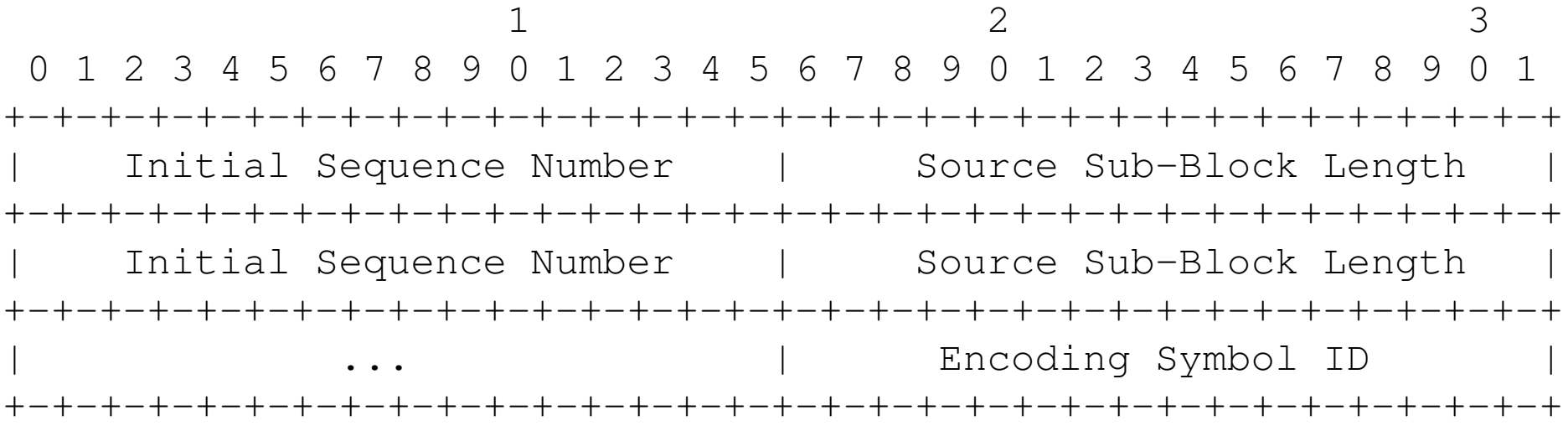
- Focus on Raptor/RaptorQ FEC code option (RFC 6681)
- Can be sufficiently general to apply to other FEC FRAME codes in current form
  - RFC 6865 Reed-Solomon
  - RFC 6816 LDPC
  - …
  - May require further study

RTP Header Extension Method

- Leverage RFC 5285 method for RTP header extensions
  - Also used by draft-ietf-mmusic-sdp-bundle-negotiation for MID
- Send explicit source FEC payload ID and repair FEC payload ID via RTP header extensions
  - Sec. 6.2 of RFC 6681
- Notes
  - Requires source stream modification (RTP header only)

No Source FEC Payload ID

- Source RTP streams unaffected
- Repair FEC Payload ID



Init. Seq. no. = RTP seq. no. for contributing flow (16 LSB's)
Order of flows can be inferred from order of m-lines in SDP or through other out-of-band signaling

Sample SDP – RTP Extensions

```
v=0
o=ali 1122334455 1122334466 IN IP4 fec.example.com
s=Raptor FEC Example
t=0 0
a=group:FEC-FR S1 S2 R1
m=video 30000 RTP/AVP 100
c=IN IP4 233.252.0.1/127
a=rtpmap:100 MP2T/90000
a=fec-source-flow: id=0
a=mid:S1
a=extmap:1 urn:ietf:params:rtp-hdrext:FEC-FR:SourceID
m=audio 10000 RTP/AVP 0 8 97
c=IN IP4 233.252.0.2/127
b=AS:200
a=rtpmap:0 PCMU/8000
a=mid:S2
a=extmap:1 urn:ietf:params:rtp-hdrext:FEC-FR:SourceID
a=fec-source-flow: id=1
m=application 30000 UDP/FEC
c=IN IP4 233.252.0.3/127
a=fec-repair-flow: encoding-id=6; fssi=Kmax:8192,T:128,P:A
a=repair-window:200ms
a=mid:R1
a=extmap:1 urn:ietf:params:rtp-hdrext:FEC-FR:RepairID
```

Sample SDP – No Source Modifications

```
v=0
o=ali 1122334455 1122334466 IN IP4 fec.example.com
s=Raptor FEC Example
t=0 0
a=group:FEC-FR S1 S2 R1
m=video 30000 RTP/AVP 100
c=IN IP4 233.252.0.1/127
a=rtpmap:100 MP2T/90000
a=fec-source-flow: id=0
a=mid:S1
m=audio 10000 RTP/AVP 0 8 97
c=IN IP4 233.252.0.2/127
b=AS:200
a=rtpmap:0 PCMU/8000
a=mid:S2
a=fec-source-flow: id=1
m=application 30000 UDP/FEC
c=IN IP4 233.252.0.3/127
a=fec-repair-flow: encoding-id=6; fssi=Kmax:8192,T:128,P:A
a=repair-window:200ms
a=mid:R1
```

Recommendations

- RTCWeb WG adopt this draft as informational spec
- Begun study to examine applicability of method to other codes beyond Raptor
  - Can repurpose draft to be more general FEC bundling solution