JSEP

IETF 91

Updates (I)

- Added new example section and removed old examples in appendix.
- Added text describing a=rtcp attribute
- Reworked handling of OfferToReceiveAudio and OfferToReceiveVideo (IETF 90)
- Reworked trickle ICE handling and its impact on m= and c= lines (DC Interim)

Updates (II)

- Added max-bundle-and-rtcp-mux policy
- Added description of maxptime handling
- Updated ICE candidate pool default to 0
- Resolved open issues around AppID/receiver-ID
- Editorial clarification

proto field in m= line (background)

- You’re supposed to use {UDP,TCP}/TLS/RTP/SAVPF
- But lots of agents try to be compatible
  - E.g., use RTP/AVPF but add a=rtcp-fb and a=fingerprint
- This is not awesome

proto field in m= line

- Offers from JSEP agents MUST be correct
- JSEP agents MUST accept
  - RTP/[S]AVP[F]
  - (UDP/TCP)/TLS/RTP/SAVP[F]
- a=fingerprint → DTLS-SRTP
- a=rtcp-fb → AVPF timing (default trr-int=0)
- No a=rtcp-fb → AVP timing (trr-int=4)
- proto field in answer MUST be equal to field in offer

ICE configuration changes

- Introduced terminology of "ICE gathering phase"
- ICE restart triggers new gathering phase
- ICE restart occurs in createOffer if:
  - ICE servers have changed
  - ICE candidate filter (e.g. relay-only -> all) changes
  - App requests ICE restart

Clarification on Bundle Policy

- The BUNDLE policy (e.g. balanced, max-compat, max-bundle) controls use of a=bundle-only
- Not whether BUNDLE is used
- JSEP endpoints always try to BUNDLE, just like they always try to rtcp-mux
- Non-JSEP endpoints can reject BUNDLE, of course

TLS/TCP for non-RTP/non-SCTP (#

R-2: DTLS-SRTP, as specified in [RFC5763], MUST be used

- Oops
- Proposed text: "*DTLS-SRTP, as specified in [RFC5763], MUST be used for media streams and DTLS MUST be used for data channels.*"

Bundle/Mux policy (#91)

- Current draft has four policies (balanced, max-compat, max-bundle, max-bundle-and-rtcp-mux)
- This seems kinda gross and non-orthogonal
- Proposal: two variables
  - bundlePolicy (balanced, max-compat, max-bundle)
  - rtcpMuxPolicy (negotiate, require)

Value of {local,remote}Description when closed (#88)

- Currently: null
- Kiran proposes: last description
- Conflict with W3C spec
- Proposal: leave-as is and fix W3C spec

a=ssrc for a=recvonly m= lines (#79)

- RTCP senders need to have a SSRC
- Currently: no a=ssrc generated, impl will pick a random SSRC and use that
- Proposal: Add a=ssrc even for a=recvonly streams
- This SSRC would be also used for RTP if the stream became sendrecv later.

Death of a one-way stream (#76)

- If remote side ends stream, no way to indicate this
- Remote SDP will change to a=sendonly, but no way to say "dead" vs "on hold"
- As such, m= line cannot be reclaimed
- IETF 89 decision to take this to MMUSIC
- Need volunteer

Multiple t= lines (#27)

- Current specification requires a t= line
  - but argues it should be a dummy (0 0)
- MT raises the question of multiple t= lines
- We don’t really use this for anything
- Proposal: clearly say that you must only send one but be silent on what you accept
- Are there other places where there should only be one something?

Signaling synchronization (#31)

- Need to indicate desire of tracks in same stream to be played in synchronized
- Proposal from DC meeting:
  - All the RTP in a given PeerConnection will use the same CNAME
  - Multiple tracks in the same stream put in the same Lip Sync (LS) group
  - LS described in https://tools.ietf.org/html/rfc5888#section-7
  - No-LS provided: All synchronized?

Changing b= (#9)

- Magnus was supposed to produce text
- New volunteer needed