

# CLUE Signalling

## Open issues and tickets

2014-11-12

Rob Hansen

# Capture IDs in RTP/RTCP (1)

- Capture ID extension to RTP/RTCP lets receivers map RTP of incoming stream to specific capture of an MCC
- Race condition when sender redefines MCC contents; problematic ones are:
  - Add a new capture to MCC
  - Convert anonymous MCC to MCC with defined sources, or vica versa
  - Convert MC to MCC with defined sources, or vica versa

# Capture IDs in RTP/RTCP (2)

- Better to have receiver deal with inconsistency than prevent sender sending new/changed media until ADVERTISEMENT has been processed
- Implementation must be able to cope with:
  - Receiving a CaptureID from an unknown Capture
  - Receiving RTP/RTCP without CaptureID when expected
  - Receiving RTP/RTCP with CaptureID when none expected
- If we didn't allow MCCs to change from anonymous to defined, or MCs to become MCCs (or vica versa) last two requirements go away
  - Cost of delayed responsiveness, media glitch too high

# Fallback from CLUE to non-CLUE

- Ticket #50
- Call only becomes non-CLUE on SDP O/A where one side removes SCTP channel from CLUE grouping semantic
  - MUST stop sending media on CLUE-controlled m-lines
  - MUST close the CLUE protocol if still enabled
- Distinct from case where CLUE channel fails
  - Media continues

# Security

- Ticket #53
- Framework document covers the basics of using SIP, SDP etc
  - Repeat, move, reference?
  - Need to also cover SCTP
- More streams means video DOS attack is worse
  - Alleviate with DTLS, ICE to confirm far end wants to receive media
- Adding feature tag, CLUE grouping semantic means more info for device fingerprinting
  - Encrypt your signalling

# Simulcast

- Ticket #47, assigned to rtp mapping, but likely to affect signalling
- If MMUSIC defines simulcast as being within a single m-line, becomes an additional tool (like BUNDLE) that CLUE can use, but is agnostic to
  - See draft-burman-mmusic-sdp-simulcast-00
- Do we continue to allow CLUE native simulcast (mapping a capture to multiple encodings)?