Media Negotiation/Transport Requirements (mostly from RTCWEB)

MMUSIC

Cullen Jennings

Justin Uberti

Eric Rescorla (*)

Desirable Properties (high priority in red)

[draft-jennings-mmusic-media-req-00]

- Push as many media flows as possible over one transport 5-tuple *
- Negotiate with both new and old endpoints without multiple O/A exchanges *
- Document how attributes behave when m-lines are merged
- Be able to independently negotiate media parameters for each flow
- Allow for a very large ($>> 100$) number of media flows
- Add new tracks without worrying about glare
- Somehow directly reference individual WebRTC tracks

* Hopefully solved by bundle/bungle/etc.

Push as many media flows as possible over one transport 5-tuple

- Priority here is *persistent* NAT bindings
  - Though minimizing initial NAT bindings is good too

Negotiate with both new and old endpoints without multiple O/A exchanges

- Or at least get media flowing with one exchange
- Might need second exchanges later

Document how attributes behave when m-lines are merged

- See Suhas's presentation

Be able to independently negotiate parameters for each flow

- Examples
  - Codecs
  - Which layers of encoding (simulcast, scalable codecs, etc.)
  - RTX
  - Resolution
- Easy if you have one m-line per flow, hard otherwise

Allow for a very large ($>> 100$) number of media flows

- Probably not displayed at one time
- Though need not be efficient when interoperating with old endpoints

Add new tracks without worrying about glare*

- Major use case here is large conference settings (e.g., hangouts)
  - New user joins; want to add their video to thumbnail
- Challenge: some of these mechanisms freak out pc.localDescription
- Is it important/valuable to do this without signaling at all?

*Not in the text of the draft.

Somehow directly reference individual WebRTC tracks

- Idea is to map flows to external information
  - E.g., participant name
- Challenge: can you use SSRCs here?

Non-goals

> “Working with SIP proxies or B2BUA that are not compliant with the standards. The reason for this is it is just not possible to design for every possible thing that does not do what the standards require.” (§4).

- This may be a tiny bit controversial