RTCWEB Items

Matthew Kaufman

1 April 2011

Chartered Work

1. Define the communication model in detail, including how session management is to occur within the model.
2. Define a security model that describes the security and privacy goals and specifies the security protocol mechanisms necessary to achieve those goals.
3. Define the solution - protocols and API requirements – for firewall and NAT traversal.
4. Define which media functions and extensions shall be supported in the client and their usage for real-time media, including media adaptation to ensure congestion safe usage.
5. Define what functionalities in the solution, such as media codecs, security algorithms, etc., can be extended and how the extensibility mechanisms works.
6. Define a set of media formats that must or should be supported by a client to improve interoperability.
7. Define how non media data is transported between clients in a secure and congestion safe way.
8. Provide W3C input for the APIs that comes from the communication model and the selected components and protocols that are part of the solution.
9. The group will consider options for interworking with legacy VoIP equipment.

- A whole lot of open questions
  - Many fall into >1 of the work categories

Communication model in detail and session management

- Web Centric
  - That is to say: Everything must be possible in a browser (might be used elsewhere, but browser limitations apply)
- SDP equivalent needed?
- Candidate addresses
  - Standard format needed?
- Peer-to-peer media, HTTP/WebSockets signalling
- Media over HTTP capability needed?
- Multi-party calling cases?
- Multiple simultaneous streams of same type cases?
  - Video+Screen-share, Front and back camera simultaneously

Security and privacy model

- User consent
  - Camera & Microphone
  - Per domain?
  - Revokable
- Protocol consent
  - STUN connectivity check?
  - Full ICE?
  - Alternatives with better legacy support?
- Protocol security
  - SRTP?
  - DTLS-SRTP?
  - End-to-end keying
  - Override of keying

Firewall and NAT traversal

- ICE?
- Subset of ICE?
- Built-in vs. script (for browser case)?
- Media relays
  - TURN?
  - Server-controlled transparent UDP relay?
  - TCP Relay?
  - Media over HTTP?

Client real-time media functions

- (S)RTP?
- (S)RTCP?
  - Extensions?
- FEC?
- RTP muxing on same port?
- Congestion control?

Extensibility mechanisms

- ?

Supported Media Formats

- Audio
  - Opus
  - G.711?
  - iLBC?
- Video
  - ?

Non-media data

- Datagrams or streams?
- Reliable delivery or not?
- In RTP or not?
- Security
- Congestion control

W3C input for the APIs

- Capability detection
- Device selection
  - Camera, Microphone
  - Separate audio for alerting?
  - Loopback (video preview, local audio testing)
  - New-device-connected events
- Codec selection and parameters
  - Which codecs
  - Rates, parameters
- Pre-processing controls
  - Echo canceller
  - AGC
  - Image stabilization
- User permissions
- Offline notification?

Interworking with legacy VoIP

- Codecs
- STUN connectivity check acceptable?
- Enough exposed to do SIP/SDP on other side?