

WebRTC Data Channels

draft-jesup-rtcweb-data-01

Randell Jesup

Salvatore Loreto

Michael Tuexen

IETF 82 Taipei

Uses

- Side channels during a 'call' (mute status, etc)
- Chat
- File transfer
- Application synchronization
- Games
- Shared whiteboard
- Co-browsing
- Shared document editing (with audio and/or video)
- Many uses we haven't thought of yet

Data Channel Requirements

- Multiple data channels
- Reliable and unreliable
- Datagram and Stream (if reliable) paradigms
- MUST be congestion-controlled
- MUST be secure (*)
- Quality open-source userland implementation needed for deployment
- See draft for other implementation requirements

Options

- Pseudo-TCP-over-UDP (reliable) + DCCP (unreliable), both over DTLS-(ICE)-UDP
 - Pseudo-TCP: no specification; in-use with source code
 - DCCP: specification; no user land implementation
- SCTP-DTLS-(ICE)-UDP or
- DTLS-SCTP-(ICE)-UDP
 - DTLS-SCTP specified (RFC 6083), SCTP-DTLS not currently (believed to be straightforward)
 - Provides reliable, unreliable, partial-reliable, datagrams and streams

Pseudo-TCP-over-UDP (reliable) + DCCP (unreliable)

- Pros
 - Well-known protocols
 - Open-source pseudo-TCP available
- Cons
 - Two protocols needed
 - Loss-based congestion control (DCCP CCID3 is similar to TFRC)
 - No known-stable user-space DCCP available
 - Multiple congestion-control flows (fights between flows)

SCTP-DTLS-(ICE)-UDP or DTLS-SCTP-(ICE)-UDP

- Pros

- Single kitchen-sink protocol
- Open-source userspace implementation based on FreeBSD
- Direct support for stream API (in SCTP-DTLS)
- Option of partial-reliability and out-of-order delivery
- Single congestion-control flow

- Cons

- Limitations sending large datagrams (but SCTP-DTLS can use streams)
- Loss-based congestion control (but replaceable)
- SCTP-DTLS has no draft currently
- Single receive window (see Open Issues)

SCTP-DTLS-(ICE)-UDP vs DTLS-SCTP-(ICE)-UDP

- **SCTP-DTLS**

- Direct use of the SCTP API
 - Such as reliable-channel streaming, partial-reliability, etc
- No draft, though should be straightforward
- Interleaving of large datagrams can (easily) be added to SCTP

- **DTLS-SCTP**

- Can use kernel implementation (browsers won't, though)
- DTLS-SCTP specified in RFC 6083.
- Reliable channels would be datagrams, not streams (or needs an extra layer)

Open issues

- SCTP
 - Michael Thornburg's issues
 - Blocking of other channels if one isn't serviced
 - Status of userland implementation derived from FreeBSD
 - Draft for SCTP-DTLS needed if chosen
 - Interleaving of large datagrams
- DCCP
 - Is a userland implementation available? Quality?
- General
 - Inter-stream priority (nice-to-have)
 - Congestion control interactions with app and media streams
 - PMTU sensing

Questions/Discussion

- Is there consensus on using SCTP?
- If so, what are people's opinions on ordering with DTLS?
 - What information is needed before consensus can be reached?