WebRTC Data Channels

Randell Jesup

IETF Interim 6/2012
Changes since last Interim

- 2-way handshake to 3-way handshake
  - Supports sending data before handshake is complete
- Updated libsctp implementation
  - Upper API updated to be generic
  - Work continues on DTLS integration
Open issues for W3

• When can data be sent on a DataChannel object?
  1) Immediately (may buffer, in-order forced until 3-way handshake complete)
  2) After the connected event

• When can we call pc.createDataChannel()?
  • Before the peerconnection is connected?
  • After the peerconnection is connected?
Proposal:

- If we call `createDataChannel` before `createOffer`, the DataChannel is queued to be connected when the PeerConnection is connected.
  
  - This will cause the offer to include an `m=` line

- If we did not have any DataChannels, and someone calls `createDataChannel`, a renegotiation may occur (to add an `m=` line).
Proposal (cont)

- If we allow `createDataChannel()` before the PeerConnection is connected, I'm mildly in favor we require waiting for the DataChannel connected event to send data. This limits issues with buffering.
- If we don't allow `createDataChannel()` before PeerConnection, I'm mildly in favor of allowing Send before the DataChannel is connected.
Early Data Pros & Cons

• Pro:
  • Faster create-to-delivered time
  • Simpler create-and-send

• Con:
  • Memory buffering
  • Less interchangeable with WebSockets