SDP negotiation of DataChannel sub-protocols

draft-ejzak-mmusic-data-channel-sdpneg-00

draft-ejzak-dispatch-msrp-data-channel-00

IETF 89

London

Keith Drage

Problem Statement

- How to negotiate use of well-defined sub-protocols over DataChannels
  - For sub-protocols that usually use SDP for negotiation, e.g., MSRP, BFCP, T140, T38
  - To support e2e signaling between different endpoint types via protocols that depend on SDP for media negotiation (e.g., SIP)
  - To allow interworking through gateways to endpoints that do not support DataChannels
  - To also support non-WebRTC endpoints
  - To support e2e negotiation of new protocols using DataChannel transport such as clue control

DataChannel external negotiation

- The proposal of this draft is that negotiation gets bolted on top of the rtcweb data channel to configure particular channels that are established using the rtcweb data channel protocol.
- Some mechanism other than the data channel control protocol is used between peers to negotiate the data channel parameters for each bi-directional data channel
  - In particular, the stream identifier is selected
- The SCTP protocol at each peer is told of the agreed data channel parameters and the DC is available for use as soon as the SCTP association is established

Stream identifier assignment rules (1)

1. For external negotiation before establishment of the SCTP association, the application can select stream ids without waiting for determination of DTLS role. Both peers need to know the stream id allocated to each data channel via the external negotiation mechanism, so deferred selection cannot be used.
2. Application should keep track of streams negotiated with external vs. DCEP so that SDP offer / answer results procedures in closing/modifying streams that were negotiated via external SDP offer / answer only.
3. Once DTLS is impacted then any DCEP associated numbers get assigned numbers.
4. Various strategies are possible to avoid conflicts, but the draft will indicate that the simplest solution is that the application assigns all the numbers to avoid complexities.

Stream identifier assignment rules (2)

1. The does not preclude other strategies being followed. More complex rules can be defined by the application as long as the rules won't result in glare and/or conflicts.
2. Application should keep track of streams negotiated with external vs. DCEP so that SDP O/A results procedures in closing/modifying streams that were negotiated via external SDP O/A only.
3. Application Needs coordinate assigning numbers to avoid conflicts. Initial offer will assign from one set of numbers, initial answer from another set of numbers; numbers must be distinct. Number set used determined by first offerer to create the SDP session.
4. Protocol stack will also follow odd even rule, application recommended to also follow this.

SDP for external negotiation

- Within the m line for the SCTP association in the SDP offer, add the following attribute(s) for each data channel
  - One attribute line to specify DC parameters (stream id, reliabile/partial-reliable, max_retr/max_time, ordered/unordered, sub-protocol)
  - Zero or more sub-protocol specific attribute lines.
- If the answer echoes the attribute with the DC parameters, the data channel is agreed. A single set of DC paremeters are negotiated for both offerer and answerer use.
- Offerer provides DC parameters to DC stack as part of DC open before sending offer. Answerer provides these after SDP offer is received.
- Sub-protocol negotiation completed with sub-protocol specific attributes in answer

Two new SDP attributes

- To represent data channel parameters equivalent to inband DC Open (DCEP)

  Example for SCTP port 5000:

  ```
  a=dcmap:5000 stream=2;max-time=10000;\
    ordered=1;partialreliable=0\
    label="channel 2"; \
    subprotocol="MSRP“
  ```

- Either max_retr (retransmit data channel) or max_time (timed data channel)present, but NOT both.
- DC parameters must be consistent with the underlying subprotocol transport needs.
- To represent data channel sub-protocol attributes (dcsa):

  ```
  a=dcsa:sctp-port:stream-id original-attribute
  ```

  Example:

  ```
  a=dcsa:5000:2 accept-types:text/plain
  ```

- Note changes from draft!

Example SDP offer

```
m=application 54111 DTLS/SCTP 5000
c=IN IP4 79.97.215.79
a=sctpmap:5000 sctp-datachannel 16
a=dcmap:5000 stream=2; \
  label="channel 2“;subprotocol="MSRP“
a=dcsa:5000:2 accept-types:text/plain
```

Note: Changes to sctpmap attribute might be appropriate but that is defined in a separate mmusic draft

Changes from previous version (in dispatch WG)

- The principal change has been to generalise the draft so it is applicable to usage over SCTP in non-rtcweb SDP usages as well as for rtcweb usages

Changes and open issues from draft

- Generic and SDP-specific stream id assignment
- Attribute names “data-channel” and “dcsa”
- Typos
- Sub-protocol registry TBD
- Need to generalize procedures for non-browser applications
- Cleanup and update needed, but generally in good shape

SDP offer/answer scenarios

- Stream id allocation odd/even rule is based on the SDP offer which creates the corresponding SCTP/DTLS association. This could be done in initial SDP offer / answer or subsequent offer / answer.
- If the DTLS/SCTP m= line is removed via setting port to 0, then a subsequent offer / answer may reuse the same m= line for new DTLS/SCTP association which may reverse the previous even/odd rules.
- SDP offer has no a=dcmap. This has two use cases:
  - Initial SDP offer: no data channel streams (DCS) are open yet.
  - Subsequent SDP offer: Remove all previously opened data channels/streams.
  - In either case, the SCTP association is kept open (and ICE procedures are performed) even when no data channels are in use.

SDP offer/answer scenarios cont..

- Creation of new DCS(s) using SDP offer (initial or subsequent)
  - Offerer has to create the data channels locally first by doing createDataChannel or equivalent API into data channel stack before sending offer.
- SDP offer was rejected
  - DCSs are kept per previous SDP offer/answer. This means DCSs that are to be deleted has to be deleted (i.e. calling createDataChannel or equivalent API) only after successful SDP answer.
  - If any new DCSs are attempted to be created per new SDP offer, these have to be closed now.

SDP offer/answer scenarios cont..

- Closing a DCS
  - Offerer should only close data channel after receiving successful SDP answer.
  - Answerer can close DCS after offer answer is received.
  - This also means offerer may receive SCTP SSN reset before SDP answer is received, which will result in data channel stack information application. Application should be aware of this and not generate a new SDP offer to close the DCS.
  - If SSN reset is received on a DCS which is not in the process of being closed (SDP answer pending or already closed by application) then initiate an SDP offer to close the stream via external means as well. May generate simultaneous offers by both peers; application need to have them (anyway).

MSRP over DataChannels

- One DC per MSRP session
- MSRP URI adds “dc” transport option
- msrp-scheme is always “MSRPS” with DC transport
- Use default DC reliability parameters (reliable, ordered) with binary format
- dcsa attributes can include: path, accept-types, accept-wrapped-types, max-size, sendonly, etc., setup, msrp-cema, [file attributes from RFC5547]

Example SDP for MSRP over DC

```
m=application 54111 DTLS/SCTP 5000
c=IN IP4 79.97.215.79
a=sctpmap:5000 webrtc-datachannel 16
a=data-channel:5000 stream=1;label="chat"; subprotocol="MSRP“
a=dcsa:5000:1 accept-types:message/cpim text/plain
a=dcsa:5000:1 path:msrps://bob.example.com:54111/si438dsaodes;dc
a=data-channel:5000 stream=2;label="file transfer"; subprotocol="MSRP“
a=dcsa:5000:2 sendonly
a=dcsa:5000:2 accept-types:message/cpim
a=dcsa:5000:2 accept-wrapped-types:*
a=dcsa:5000:2 path:msrps://bob.example.com:54111/jshA7we;dc
a=dcsa:5000:2 file-selector:name:"My cool picture.jpg" \
           type:image/jpeg size:32349 hash:sha-1: \
           72:24:5F:E8:65:3D:DA:F3:71:36:2F:86:D4:71:91:3E:E4:A2:CE:2E
a=dcsa:5000:2 file-transfer-id:vBnG916bdberum2fFEABR1FR3ExZMUrd
a=dcsa:5000:2 file-disposition:attachment
a=dcsa:5000:2 file-date:creation:"Mon, 15 May 2006 15:01:31 +0300„
a=dcsa:5000:2 file-icon:cid:id2@bob.example.com
a=dcsa:5000:2 file-range:1-32349
```

MSRP configurations supported

- e2e MSRP over data channel(s)
  - As described
- Via gateway to TLS/TCP transport
  - Using CEMA or MSRP B2BUA to interwork MSRP over DC with MSRP over TLS/TCP
  - MSRP transport related attributes (a=setup) and related semantics do not apply when data channel is used as transport.

Example for CLUE

- CLUE subprotcol mappings

```
a= a=dcmap:5000 stream=3;\
   ordered=1;partialreliable=0\
   label=“CLUE channel"; \
   subprotocol=“CLUE“
a=dcsa:5000:3 <CLUE attributes from [I-D.kyzivat-clue-signaling]>
```

- “CLUE” protocol name yet to be added to the 'Protocol Registry' defined by [I-D.ietf-rtcweb-data-protocol]. This is expected to be defined by [I-D.ietf-holmberg-clue-datachannel].

Proposed work plan

- draft-ejzak-mmusic-data-channel-sdpneg-00
- draft-ejzak-dispatch-msrp-data-channel-00
- One draft for each additional protocol to be supported:
  - BFCP, T140, clue control, (T38?, others?)