CLUE WG R. Even, Ed.
Internet-Draft Huawei Technologies
Intended status: Informational June 22, 2014
Expires: December 24, 2014

CLUE protocol Call Flows
draft-even-clue-call-flows-00.txt

Abstract

This document provides some call flows examples using the CLUE extensions for "telepresence"

Status of This Memo

This Internet-Draft will expire on December 24, 2014.

Copyright Notice

Table of Contents

1. Introduction

This document provides some call flows examples using the CLUE extensions for "telepresence". The examples include a typical point to point call between two endpoint with three cameras and screens. A call from a telepresence endpoint to an endpoint that do not support the CLUE telepresence extensions. An point to point call between a three screens and three camera endpoint to a single screen and single camera end point both support the CLUE telepresence extensions.

The examples will not include ICE and SRTP negotiations but the actual usage SHOULD include ICE and SRTP.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC2119[RFC2119] and indicate requirement levels for compliant RTP implementations.

3. Symmetric point to point Telepresence call

In this example both end points have three monitors and three cameras and fully support the CLUE telepresence extensions.

The initial call is from Alice to Bob. The first offer includes an audio and video channel, a data channel for CLUE and the CLUE feature tag.

```
INVITE sip:bob@biloxi.example.com SIP/2.0

Via: SIP/2.0/TCP
client.atlanta.example.com:5060;branch=z9hG4bK74bf9

Max-Forwards: 70

From: Alice <sip:alice@atlanta.example.com>;tag=9fxced76sl
```

Call-ID: 3848276298220188511@atlanta.example.com

CSeq: 1 INVITE

Contact: sip:alice@client.atlanta.example.com;transport=tcp; CLUE (?)

Content-Type: application/sdp

Content-Length: xxx

v=0

o=alice 2890844526 2890844526 IN IP4 client.atlanta.example.com

s=-

c=IN IP4 192.0.2.101

t=0 0

m=audio 49172 RTP/AVP 0

a=rtpmap:0 PCMU/8000

m=video 49174 RTP/AVP 96

a=rtpmap:96 H264/90000

a=fmtp:96 profile-level-id=42e016;max-mbps=108000;max-fs=3600

a=sendrecv

m=application 54111 DTLS/SCTP 54111

a=sctpmap:54111 webrtc-datachannel

SIP/2.0 200 OK

Via: SIP/2.0/TCP client.atlanta.example.com:5060;branch=z9hG4bK74bf9

;received=192.0.2.101

From: Alice <sip:alice@atlanta.example.com>;tag=9fxced76sl

To: Bob <sip:bob@biloxi.example.com>;tag=8321234356

```
Call-ID: 3848276298220188511@atlanta.example.com

CSeq: 1 INVITE

Contact: sip:bob@client.biloxi.example.com;transport=tcp; CLUE (?)

Content-Type: application/sdp

Content-Length: zzz

v=0

o=bob 2890844527 2890844527 IN IP4 client.biloxi.example.com

s=-

c=IN IP4 192.0.2.201

t=0 0

m=audio 3456 RTP/AVP 0

a=rtpmap:0 PCMU/8000

m=video 3458 RTP/AVP 96

a=rtpmap:96 H264/90000

a=fmtp:96 profile-level-id=42e016;max-mbps=108000;max-fs=3600

a=sendrecv

m=application 54100 DTLS/SCTP 54111

a=sctpmap:54100 webrtc-datachannel

ACK sip:bob@client.biloxi.example.com SIP/2.0

Via: SIP/2.0/TCP
client.atlanta.example.com:5060;branch=z9hG4bK74bd5

Max-Forwards: 70

From: Alice <sip:alice@atlanta.example.com>;tag=9fxced76sl

To: Bob <sip:bob@biloxi.example.com>;tag=8321234356

Call-ID: 3848276298220188511@atlanta.example.com
```

CSeq: 1 ACK

Content-Length: 0

After establishing the initial SIP connection Alice and Bob need to open the CLUE channel.

The CLUE data channel is based on the RTCweb data channel as specified in http://tools.ietf.org/html/draft-ietf-clue-datachannel-00.

The first step is to open the DTLS [RFC6347] connection . The DTLS connection will be opened by Alice

```
 Alice                                                Bob

      ClientHello                  -------->
                                                   ServerHello
                                                  Certificate*
                                            ServerKeyExchange*
                                           CertificateRequest*
                                   <--------   ServerHelloDone
      Certificate*
      ClientKeyExchange
      CertificateVerify*
      [ChangeCipherSpec]
      Finished                     -------->
                                            [ChangeCipherSpec]
                                   <--------          Finished
      Application Data             <------->  Application Data
```

After establishing the DTLS connection the SCTP association is created as specified in [RFC4960]. The INIT and INITACK include the number of channels that will be used.

```
   Alice (A)                                         Bob (Z)

       INIT [I-Tag=Tag_A OS=1 MIS=1
            I-TSN=0 & other info] ------>
                                          INIT ACK [Veri Tag=Tag_A,
                                                 I-Tag=Tag_Z,
                                 <------   Cookie_Z, & other info]

       COOKIE ECHO [Cookie_Z] ---->
                                     <----   COOKIE-ACK
```

The SCTP messages are carried in the DATA messages.

The next step is to open a web RTC channel [I-D.ietf-rtcweb-data-protocol]. PPID 50 is the webRTC Data Channel Establishment Protocol (DCEP) [I-D.ietf-rtcweb-data-protocol]]. PPID 51 is the CLUE protocol ID [I-D.ietf-clue-datachannel].

The SCTP DATA message is as follows, the Stream Sequence number will progress.

```
   DATA [TSN=initial TSN=0
           Strm=0,Seq=0
           ppid= 50; & user data]--------->          SACK [TSN = 0,
                                            <-----------   Block=0]
```

The first SCTP data message from Alice will carry the DATA_CHANNEL_OPEN message. This will open a bi-directional channel. DATA_CHANNEL_OPEN [message type=3, DATA_CHANNEL_RELIABLE, Label Length = 0, Protocol Length = 4, protocol=CLUE)

Bob Answers with DATA_CHANNEL_ACK [message type=2]

The next SCTP DATA messages will use PPID = 51 since it will carry the CLUE protocol. The Clue Exchange starts from Alice

Question: do we want full XML for CLUE messages or just pseudo code providing the parameters?

```
Alice                              Bob
Option [sequenceNr=1,
media provider=true,
media consumer=true]. ----------------------------->
                                                       OptionResponse
                                                       [sequenceNr=4
                                                       ResponseCode,
                                                       ResponseString,
                                                       media provider=true,
                                   <--------------------   media consumer=true].
```

Alice sends an advertisement to Bob, Alice will also send a new SIP invite with the sendonly CLUE media streams. The SIP call flow is in section 7 of [I-D.ietf-clue-signaling] (should it be moved here?)

```
Advertisement [sequenceNr =2,
mediacaptures,
encodinggroups,
captureScenes] -------------------->
```

Bob can now send a Configure message asking for the three cameras and video, a SIP message that will specify receive only RTP streams for the m-lines in the offer from ALICE with sendonly streams . The advertisement acknowledge to Alice is in the configure message.

Bob will also send an Advertisement and a SIP INVITE with the send only RTP media streams.

```
                                                Configure [sequenceNr=6,
                                                          advsequenceNr=2
                                                       ack=true
                                        <-------------- captureEncodings]
Configue Response [sequenceNr=3,
                 ResponseCode,
                ResponseString,
               confSequenceNr=6]----------->


                                                        Advertisement
                                                        [sequenceNr =7,
                                                        mediacaptures,
                                                        encodinggroups,
                                     <---------------- captureScenes]
```

Alice will now send the CONFIGURE message and the SIP Invite for receiving the send only RTP streams from Bob

```
Configure [sequenceNr=4,
advsequenceNr=7
ack=true
captureEncodings] -------------------------------->

                                                    Configue Response
                                                     [sequenceNr=8,
                                                     ResponseCode,
                                                     ResponseString,
                                    <-------------  confSequenceNr=4]
```

4. Acknowledgements

5. IANA Considerations

This document contains no IANA considerations.

6. Security Considerations

While there are likely to be security considerations for any solution for telepresence , this document has no security considerations.

7. Informative References

[I-D.ietf-clue-data-model-schema]
Presta, R. and S. Romano, "An XML Schema for the CLUE data model", draft-ietf-clue-data-model-schema-05 (work in progress), June 2014.

[I-D.ietf-clue-datachannel]
Holmberg, C., "CLUE Protocol Data Channel", draft-ietf-clue-datachannel-00 (work in progress), March 2014.

[I-D.ietf-clue-protocol]
Presta, R. and S. Romano, "CLUE protocol", draft-ietf-clue-protocol-00 (work in progress), June 2014.

[I-D.ietf-clue-signaling]
Kyzivat, P., Xiao, L., Groves, C., and R. Hansen, "CLUE Signaling", draft-ietf-clue-signaling-01 (work in progress), May 2014.

[I-D.ietf-rtcweb-data-protocol]
Jesup, R., Loreto, S., and M. Tuexen, "WebRTC Data Channel Establishment Protocol", draft-ietf-rtcweb-data-protocol-06 (work in progress), June 2014.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3265] Roach, A., "Session Initiation Protocol (SIP)-Specific Event Notification", RFC 3265, June 2002.

[RFC4960] Stewart, R., "Stream Control Transmission Protocol", RFC 4960, September 2007.

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, January 2012.

Author’s Address

Roni Even (editor)
Huawei Technologies
Tel Aviv
Israel

Email: roni.even@mail01.huawei.com