CLUE WG   R. Even
Internet-Draft   Huawei Technologies
Intended status: Standards Track   J. Lennox
Expires: April 20, 2016   Vidyo
                                                       October 18, 2015

Mapping RTP streams to CLUE media captures
draft-ietf-clue-rtp-mapping-05.txt

Abstract

This document describes how the Real Time transport Protocol (RTP) is used in the context of the CLUE protocol. It also describes the mechanisms and recommended practice for mapping RTP media streams defined in SDP to CLUE media captures.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Telepresence systems can send and receive multiple media streams. The CLUE framework [I-D.ietf-clue-framework] defines media captures as a source of Media, such as from one or more Capture Devices. A Media Capture may also be constructed from other Media streams. A middle box can express conceptual Media Captures that it constructs from Media streams it receives. A Multiple Content Capture (MCC) is a special Media Capture composed of multiple Media Captures.

SIP offer answer [RFC3264] uses SDP [RFC4566] to describe the RTP[RFC3550] media streams. Each RTP stream has a unique SSRC within its RTP session. The content of the RTP stream is created by an encoder in the endpoint. This may be an original content from a camera or a content created by an intermediary device like an MCU.

This document makes recommendations, for the telepresence architecture, about how RTP and RTCP streams should be encoded and transmitted, and how their relation to CLUE Media Captures should be communicated. The proposed solution supports multiple RTP topologies.

With regards to the media (audio and video), systems that support CLUE use RTP for the media, SDP for codec and media transport negotiation (CLUE individual encodings) and the CLUE protocol for media Capture description and selection. In order to associate the media in the different protocols there are three mapping that need to be specified:

1. CLUE individual encodings to SDP

2. RTP media streams to SDP (this is not a CLUE specific mapping)

3. RTP media streams to MC to map the received RTP steam to the current MC in the MCC.

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. RTP topologies for CLUE

The typical RTP topologies used by Telepresence systems specify different behaviors for RTP and RTCP distribution. A number of RTP topologies are described in [I-D.ietf-avtcore-rtp-topologies-update]. For telepresence, the relevant topologies include point-to-point, as well as media mixers, media- switching mixers, and Selective Forwarding middleboxs.

In the point-to-point topology, one peer communicates directly with a single peer over unicast. There can be one or more RTP sessions, and each RTP session can carry multiple RTP streams identified by their SSRC. All SSRCs will be recognized by the peers based on the information in the RTCP SDES report that will include the CNAME and SSRC of the sent RTP streams. There are different point to point use cases as specified in CLUE use case [RFC7205]. There may be a difference between the symmetric and asymmetric use cases. While in the symmetric use case the typical mapping will be from a Media capture device to a render device (e.g. camera to monitor) in the asymmetric case the render device may receive different capture information (RTP stream from different cameras) if it has fewer rendering devices (monitors). In some cases, a CLUE session which, at a high-level, is point-to-point may nonetheless have RTP which is best described by one of the mixer topologies. For example, a CLUE endpoint can produce composite or switched captures for use by a receiving system with fewer displays than the sender has cameras. The Media capture may be described using MCC.

For the Media Mixer topology [I-D.ietf-avtcore-rtp-topologies-update], the peers communicate only with the mixer. The mixer provides mixed or composited media streams, using its own SSRC for the sent streams. There are two cases here. In the first case the mixer may have separate RTP sessions with each peer (similar to the point to point topology) terminating the RTCP sessions on the mixer; this is known as Topo-RTCP-Terminating MCU in [I-D.ietf-avtcore-rtp-topologies-update]. In the second case, the mixer can use a conference-wide RTP session similar to [I-D.ietf-avtcore-rtp-topologies-update] Topo-mixer or Topo-Video-switching. The major difference is that for the second case, the mixer uses conference-wide RTP sessions, and distributes the RTCP reports to all the RTP session participants, enabling them to learn all the CNAMEs and SSRCs of the participants and know the contributing source or sources (CSRCs) of the original streams from the RTP header. In the first case, the Mixer terminates the RTCP and the participants cannot know all the available sources based on the RTCP information. The conference roster information including conference participants, endpoints, media and media-id (SSRC) can be available using the conference event package [RFC4575] element.

In the Media-Switching Mixer topology [I-D.ietf-avtcore-rtp-topologies-update], the peer to mixer communication is unicast with mixer RTCP feedback. It is conceptually similar to a compositing mixer as described in the previous paragraph, except that rather than compositing or mixing multiple sources, the mixer provides one or more conceptual sources selecting one source at a time from the original sources. The Mixer creates a conference-wide RTP session by sharing remote SSRC values as CSRCs to all conference participants.

In the Selective Forwarding middlebox topology, the peer to mixer communication is unicast with RTCP mixer feedback. Every potential sender in the conference has a source which may be "projected" by the mixer into every other RTP session in the conference; thus, every original source is maintained with an independent RTP identity to every receiver, maintaining separate decoding state and its original RTCP SDES information. However, RTCP is terminated at the mixer, which might also perform reliability, repair, rate adaptation, or transcoding on the stream. Senders' SSRCs may be renumbered by the mixer. The sender may turn the projected sources on and off at any time, depending on which sources it thinks are most relevant for the receiver; this is the primary reason why this topology must act as an RTP mixer rather than as a translator, as otherwise these disabled sources would appear to have enormous packet loss. Source switching is accomplished through this process of enabling and disabling projected sources, with the higher-level semantic assignment of reason for the RTP streams assigned externally.

The above topologies demonstrate two major RTP/RTCP behaviors:

1. The mixer may either use the source SSRC when forwarding RTP packets, or use its own created SSRC. Still the mixer will distribute all RTCP information to all participants creating conference-wide RTP session/s. This allows the participants to learn the available RTP sources in each RTP session. The original source information will be the SSRC or in the CSRC depending on the topology. The point to point case behaves like this.

2. The mixer terminates the RTCP from the source, creating separate RTP sessions with the peers. In this case the participants will not receive the source SSRC in the CSRC. Since this is usually a mixer topology, the source information is available from the SIP conference event package [RFC4575]. Subscribing to the conference event package allows each participant to know the SSRCs of all sources in the conference.

4. Mapping CLUE Capture Encodings to RTP streams

The different topologies described in Section 3 create different SSRC distribution models and RTP stream multiplexing points.

Most video conferencing systems today can separate multiple RTP sources by placing them into separate RTP sessions using, the SDP description. For example, main and slides video sources are separated into separate RTP sessions based on the content attribute [RFC4796]. This solution works straightforward if the multiplexing point is at the UDP transport level, where each RTP stream uses a separate RTP session. This will also be true for mapping the RTP streams to Media Captures Encodings if each media capture encodings uses a separate RTP session, and the consumer can identify it based on the receiving RTP port. In this case, SDP only needs to label the RTP session with an identifier that can be used to identify the media capture in the CLUE description. The SDP label attribute serves as this identifier. In this case, the mapping does not change even if the RTP session is switched using same or different SSRC. (The multiplexing is not at the SSRC level).

Even though Session multiplexing is supported by CLUE, for scaling reasons, CLUE recommends using SSRC multiplexing in a single or multiple sessions using [I-D.ietf-mmusic-sdp-bundle-negotiation]. So we need to look at how to map RTP streams to Captures Encodings when SSRC multiplexing is used.

When looking at SSRC multiplexing we can see that in various topologies, the SSRC behavior may be different:

1. The SSRCs are static (assigned by the MCU/Mixer), and there is an SSRC for each media capture encoding defined in the CLUE protocol. Source information may be conveyed using CSRC, or, in the case of topo-RTCP-Terminating MCU, is not conveyed.

2. The SSRCs are dynamic, representing the original source and are relayed by the Mixer/MCU to the participants.

In the above two cases the MCU/Mixer may create an advertisement, with a virtual room capture scene.

Another case we can envision is that the MCU / Mixer relays all the capture scenes from all advertisements to all consumers. This means that the advertisement will include multiple capture scenes, each representing a separate TelePresence room with its own coordinate system.

MCCs bring another mapping issue, in that an MCC represents multiple Media Captures that can be sent as part of this MCC if configured by the consumer. When receiving an RTP stream which is mapped to the MCC, the consumer needs to know which original MC it is in order to get the MC parameters from the advertisement. If a consumer requested a MCC, the original MC does not have a capture encoding, so it cannot be associated with an m-line using a label as described in CLUE signaling [I-D.ietf-clue-signaling]. This is important, for example, to get correct scaling information for the original MC, which may be different for the various MCs that are contributing to the MCC.

4.1. Review of RTP related documents relevant to CLUE work.

Editor’s note: This section provides an overview of the RFCs and drafts that are can be used in a CLUE system and as a base for a mapping solution. This section is for information only; the normative behavior is given in the cited documents. Tools for SSRC multiplexing support are defined for general conferencing applications; CLUE systems use the same tools.

When looking at the available tools based on current work in MMUSIC, AVTcore and AVText for supporting SSRC multiplexing the following documents are considered to be relevant.

Negotiating Media Multiplexing Using the Session Description Protocol in [I-D.ietf-mmusic-sdp-bundle-negotiation] defines a "bundle" SDP grouping extension that can be used with SDP Offer/Answer mechanism to negotiate the usage of a single 5-tuple for sending and receiving media associated with multiple SDP media descriptions ("m="). [bundle] specifies how to associate a received RTP stream with the m-line describing it. The assumption in the work is that each SDP m-line represents a single media source. [I-D.ietf-mmusic-sdp-bundle-negotiation] specifies using the SDP mid value and sending it as RTCP SDES and an RTP header extension in order to be able to map the RTP stream to the SDP m-line. This is relevant when there are multiple RTP streams with the same payload subtype number.

SDP Source attribute [RFC5576] mechanisms to describe specific attributes of RTP sources based on their SSRC.

Negotiation of generic image attributes in SDP [RFC6236] provides the means to negotiate the image size. The image attribute can be used to offer different image parameters like size but in order to offer multiple RTP streams with different resolutions it does it using separate RTP session for each image option ([I-D.ietf-mmusic-sdp-bundle-negotiation] provides the support of a single RTP session but each image option will need a separate SDP m-line).

The recommended support of the simulcast case is to use [I-D.ietf-mmusic-sdp-simulcast]

In the next sections, the document will propose mechanisms to map the RTP streams to media captures addressing.

4.2. Requirements of a solution

This section lists, more briefly, the requirements a media architecture for Clue telepresence needs to achieve, summarizing the discussion of previous sections. In this section, RFC 2119 [RFC2119] language refers to requirements on a solution, not an implementation; thus, requirements keywords are not written in capital letters.

Media-1: It must not be necessary for a Clue session to use more than a single transport flow for transport of a given media type (video or audio).

Media-2: It must, however, be possible for a Clue session to use multiple transport flows for a given media type where it is considered valuable (for example, for distributed media, or differential quality-of-service).

Media-3: It must be possible for a Clue endpoint or MCU to simultaneously send sources corresponding to static captures and to both composited and switched multi-content captures in the same transport flow. (Any given device might not necessarily be able send all of these source types; but for those that can, it must be possible for them to be sent simultaneously.)

Media-4: It must be possible for an original source to move among multi-content captures (i.e. at one time be sent for one MCC, and at a later time be sent for another one).

Media-5: It must be possible for a source to be placed into a MCC even if the source is a "late joiner", i.e. was added to the conference after the receiver requested the MCC.

Media-6: Whenever a given source is assigned to a switched capture, it must be immediately possible for a receiver to determine the MCC it corresponds to, and thus that any previous source is no longer being mapped to that switched capture.

Media-7: It must be possible for a receiver to identify the original capture(s) that are currently being mapped to an MCC, and correlate it with both the Clue advertisement and out-of-band (non-Clue) information such as rosters.

Media-8: It must be possible for a source to move among MCCs without requiring a refresh of decoder state (e.g., for video, a fresh I-frame), when this is unnecessary. However, it must also be possible for a receiver to indicate when a refresh of decoder state is in fact necessary.

Media-9: If a given source is being sent on the same transport flow for more than one reason (e.g. if it corresponds to more than one switched capture at once, or to a static capture), it should be possible for a sender to send only one copy of the source.

Media-10: On the network, media flows should, as much as possible, look and behave like currently-defined usages of existing protocols; established semantics of existing protocols must not be redefined.

Media-11: The solution should seek to minimize the processing burden for boxes that distribute media to decoding hardware.

Media-12: If multiple sources from a single synchronization context are being sent simultaneously, it must be possible for a receiver to associate and synchronize them properly, even for sources that are are mapped to switched captures.

4.3. Static Mapping

Static mapping is widely used in current MCU implementations. It is also common for a point to point symmetric use case when both endpoints have the same capabilities. For capture encodings with static SSRCs, it is most straightforward to indicate this mapping outside the media stream, in the CLUE or SDP signaling. When using SSRC multiplexing [I-D.ietf-mmusic-sdp-bundle-negotiation] defines the use of the SDP mid attribute value to associate between the received RTP stream and the SDP m-line. The mid is carried as an RTP header extension and RTCP SDES message defined in [I-D.ietf-mmusic-sdp-bundle-negotiation] .

4.4. Dynamic mapping

Dynamic mapping by tagging each media packet with the SDP mid value. This means that a receiver immediately knows how to interpret received media, even when an unknown SSRC is seen. As long as the media carries a known mid, it can be assumed that this media stream will replace the stream currently being received with that mid.

This gives significant advantages to switching latency, as a switch between sources can be achieved without any form of negotiation with the receiver.

However, the disadvantage in using a mid in the stream that it introduces additional processing costs for every media packet, as mid are scoped only within one hop (i.e., within a cascaded conference a mid that is used from the source to the first MCU is not meaningful between two MCUs, or between an MCU and a receiver), and so they may need to be added or modified at every stage.

An additional issue with putting mid in the RTP packets comes from cases where a non-bundle aware endpoint is being switched by an MCU to a bundle endpoint. In this case, we may require up to an additional 12 bytes in the RTP header, which may push a media packet over the MTU. However, as the MTU on either side of the switch may not match, it is possible that this could happen even without adding extra data into the RTP packet. The 12 additional bytes per packet could also be a significant bandwidth increase in the case of very low bandwidth audio codecs.

4.5. Recommendations

The recommendation is that CLUE endpoint using SSRC multiplexing MUST support [[I-D.ietf-mmusic-sdp-bundle-negotiation] and use the SDP mid attribute for mapping.

5. Application to CLUE Media Requirements

The requirement section Section 4.2 offers a number of requirements that are believed to be necessary for a CLUE RTP mapping. The solutions described in this document are believed to meet these requirements, though some of them are only possible for some of the topologies. (Since the requirements are generally of the form "it must be possible for a sender to do something", this is adequate; a sender which wishes to perform that action needs to choose a topology which allows the behavior it wants.

In this section we address only those requirements where the topologies or the association mechanisms treat the requirements differently.

Media-4: It must be possible for an original source to move among switched captures (i.e. at one time be sent for one switched capture, and at a later time be sent for another one).

This applies naturally for static sources with a Switched Mixer. For dynamic sources with a Selective Forwarding middlebox, this just requires the mid in the header extension element to be updated appropriately.

Media-6: Whenever a given source is transmitted for a switched capture, it must be immediately possible for a receiver to determine the switched capture it corresponds to, and thus that any previous source is no longer being mapped to that switched capture.

For a Switched Mixer, this applies naturally. For a Selective Forwarding middlebox, this is done based on the mid.

Media-7: It must be possible for a receiver to identify the original source that is currently being mapped to a switched capture, and correlate it with out-of-band (non-Clue) information such as rosters.

For a Switched Mixer, this is done based on the CSRC, if the mixer is providing CSRCs; For a Selective Forwarding middlebox, this is done based on the SSRC.

For MCC which can represent multiple switched MCs there is a need to know which MC represents the current RTP stream, requires a mapping from an RTP stream to an MC. In order to address this mapping this document defines an RTP header extension that includes the CaptureID in order to map to the original MC allowing the consumer to use the MC attributes like the spatial information.

Media-8: It must be possible for a source to move among switched captures without requiring a refresh of decoder state (e.g., for video, a fresh I-frame), when this is unnecessary. However, it must also be possible for a receiver to indicate when a refresh of decoder state is in fact necessary.

This can be done by a Selective Forwarding middlebox, but not by a Switching Mixer. The last requirement can be accomplished through an FIR message [RFC5104], though potentially a faster mechanism (not requiring a round-trip time from the receiver) would be preferable.

Media-9: If a given source is being sent on the same transport flow to satisfy more than one capture (e.g. if it corresponds to more than one switched capture at once, or to a static capture as well as a switched capture), it should be possible for a sender to send only one copy of the source.

For a Selective Forwarding middlebox, this may be a problem since an encoding can be used by a single MC, it will require using the same SDP label for multiple MC (example middle camera and active speaker MC) this can also be done for an environment with a hybrid of mixer topologies and static and dynamic captures. It is not possible for static captures from a Switched Mixer.

Media-12: If multiple sources from a single synchronization context are being sent simultaneously, it must be possible for a receiver to associate and synchronize them properly, even for sources that are mapped to switched captures.

For a Mixed or Switched Mixer topology, receivers will see only a single synchronization context (CNAME), corresponding to the mixer. For a Selective Forwarding middlebox, separate projecting sources keep separate synchronization contexts based on their original CNAMEs, thus allowing independent synchronization of sources from independent rooms without needing global synchronization. In hybrid cases, however (e.g. if audio is mixed), all sources which need to be synchronized with the mixed audio must get the same CNAME (and thus a mixer-provided timebase) as the mixed audio.

6. CaptureID definition

For mapping an RTP stream to a specific MC in the MCC the CLUE captureId is used. The media sender MUST send for MCC the captureID in the RTP header and as a RTCP SDES message.

6.1. RTCP CaptureId SDES Item

This document specifies a new RTCP SDES message

```
 0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | CaptureId = XXX  |     length    |CaptureId
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |   ....
```

This CaptureID is the same as in the CLUE MC and is also used in the RTP header extension.

This SDES message MAY be sent in a compound RTCP packet based on the application need.

6.2. RTP Header Extension

The CaptureId is carried within the RTP header extension field, using [RFC5285] two bytes header extension.

Support is negotiated within the SDP, i.e.

a=extmap:1 urn:ietf:params:rtp-hdrext:CaptureId

Packets tagged by the sender with the CapturId then contain a header extension as shown below

```
 0                   1                   2                   3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |  ID               |   Len-1               |    CaptureId
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |  CaptureId ..        |
     +-+-+-+-+-+-+-+-+-+
```

There is no need to send the CaptureId header extension with all RTP packets. Senders MAY choose to send it only when a new MC is sent. If such a mode is being used, the header extension SHOULD be sent in the first few RTP packets to reduce the risk of losing it due to packet loss.

7. Examples

TBD

8. Acknowledgements

The authors would like to thanks Allyn Romanow and Paul Witty for contributing text to this work.

9. IANA Considerations

This document defines a new extension URI in the RTP Compact Header Extensions subregistry of the Real-Time Transport Protocol (RTP) Parameters registry, according to the following data:

Extension URI: urn:ietf:params:rtp-hdrext:CaptureId

Description: CLUE CaptureId

Contact: roni.even@mail01.huawei.com

Reference: RFC XXXX

The IANA is requested to register one new RTCP SDES items in the "RTCP SDES Item Types" registry, as follows:

Value	Abbrev	Name	Reference
TBA	CCID	CLUE CaptureId	[RFCXXXX]

10. Security Considerations

The security considerations of the RTP specification, the RTP/SAVPF profile, and the various RTP/RTCP extensions and RTP payload formats that form the complete protocol suite described in this memo apply. It is not believed there are any new security considerations resulting from the combination of these various protocol extensions.

The Extended Secure RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback [RFC5124] (RTP/SAVPF) provides handling of fundamental issues by offering confidentiality, integrity and partial source authentication. A mandatory to support media security solution is created by combining this secured RTP profile and DTLS-SRTP keying [RFC5764]

RTCP packets convey a Canonical Name (CNAME) identifier that is used to associate RTP packet streams that need to be synchronised across related RTP sessions. Inappropriate choice of CNAME values can be a privacy concern, since long-term persistent CNAME identifiers can be used to track users across multiple calls. This memo mandates generation of short-term persistent RTCP CNAMES, as specified in RFC7022 [RFC7022], resulting in untraceable CNAME values that alleviate this risk.

Some potential denial of service attacks exist if the RTCP reporting interval is configured to an inappropriate value. This could be done by configuring the RTCP bandwidth fraction to an excessively large or small value using the SDP "b=RR:" or "b=RS:" lines [RFC3556], or some similar mechanism, or by choosing an excessively large or small value for the RTP/AVPF minimal receiver report interval (if using SDP, this is the "a=rtcp-fb:... trr-int" parameter) [RFC4585] The risks are as follows:

1. the RTCP bandwidth could be configured to make the regular reporting interval so large that effective congestion control cannot be maintained, potentially leading to denial of service due to congestion caused by the media traffic;

2. the RTCP interval could be configured to a very small value, causing endpoints to generate high rate RTCP traffic, potentially leading to denial of service due to the non-congestion controlled RTCP traffic; and

3. RTCP parameters could be configured differently for each endpoint, with some of the endpoints using a large reporting interval and some using a smaller interval, leading to denial of service due to premature participant timeouts due to mismatched timeout periods which are based on the reporting interval (this is a particular concern if endpoints use a small but non-zero value for the RTP/AVPF minimal receiver report interval (trr-int) [RFC4585], as discussed in [I-D.ietf-avtcore-rtp-multi-stream]).

Premature participant timeout can be avoided by using the fixed (non-reduced) minimum interval when calculating the participant timeout ([I-D.ietf-avtcore-rtp-multi-stream]). To address the other concerns, endpoints SHOULD ignore parameters that configure the RTCP reporting interval to be significantly longer than the default five second interval specified in [RFC3550] (unless the media data rate is so low that the longer reporting interval roughly corresponds to 5% of the media data rate), or that configure the RTCP reporting interval small enough that the RTCP bandwidth would exceed the media bandwidth.

The guidelines in [RFC6562] apply when using variable bit rate (VBR) audio codecs such as Opus. The use of the encryption of the header extensions are RECOMMENDED, unless there are known reasons, like RTP middleboxes performing voice activity based source selection or third party monitoring that will greatly benefit from the information, and this has been expressed using API or signalling. If further evidence are produced to show that information leakage is significant from audio level indications, then use of encryption needs to be mandated at that time.

In multi-party communication scenarios using RTP Middleboxes, a lot of trust is placed on these middleboxes to preserve the sessions security. The middlebox needs to maintain the confidentiality, integrity and perform source authentication. The middlebox can perform checks that prevents any endpoint participating in a conference to impersonate another. Some additional security considerations regarding multi-party topologies can be found in [I-D.ietf-avtcore-rtp-topologies-update]

11. References

11.1. Normative References

[I-D.ietf-clue-framework]
Duckworth, M., Pepperell, A., and S. Wenger, "Framework for Telepresence Multi-Streams", draft-ietf-clue-framework-23 (work in progress), September 2015.

[I-D.ietf-mmusic-sdp-bundle-negotiation]
Holmberg, C., Alvestrand, H., and C. Jennings, "Negotiating Media Multiplexing Using the Session Description Protocol (SDP)", draft-ietf-mmusic-sdp-bundle-negotiation-23 (work in progress), July 2015.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

11.2. Informative References

[I-D.ietf-avtcore-rtp-multi-stream]
Lennox, J., Westerlund, M., Wu, W., and C. Perkins, "Sending Multiple Media Streams in a Single RTP Session", draft-ietf-avtcore-rtp-multi-stream-09 (work in progress), September 2015.

[I-D.ietf-avtcore-rtp-topologies-update]
Westerlund, M. and S. Wenger, "RTP Topologies", draft-ietf-avtcore-rtp-topologies-update-10 (work in progress), July 2015.

[I-D.ietf-clue-signaling]
Kyzivat, P., Xiao, L., Groves, C., and R. Hansen, "CLUE Signaling", draft-ietf-clue-signaling-06 (work in progress), August 2015.

[I-D.ietf-mmusic-sdp-simulcast]
Westerlund, M., Nandakumar, S., and M. Zanaty, "Using Simulcast in SDP and RTP Sessions", draft-ietf-mmusic-sdp-simulcast-02 (work in progress), October 2015.

[RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, DOI 10.17487/RFC3264, June 2002, <http://www.rfc-editor.org/info/rfc3264>.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, DOI 10.17487/RFC3550, July 2003, <http://www.rfc-editor.org/info/rfc3550>.

[RFC3556] Casner, S., "Session Description Protocol (SDP) Bandwidth Modifiers for RTP Control Protocol (RTCP) Bandwidth", RFC 3556, DOI 10.17487/RFC3556, July 2003, <http://www.rfc-editor.org/info/rfc3556>.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, DOI 10.17487/RFC4566, July 2006, <http://www.rfc-editor.org/info/rfc4566>.

[RFC4575] Rosenberg, J., Schulzrinne, H., and O. Levin, Ed., "A Session Initiation Protocol (SIP) Event Package for Conference State", RFC 4575, DOI 10.17487/RFC4575, August 2006, <http://www.rfc-editor.org/info/rfc4575>.

[RFC4585] Ott, J., Wenger, S., Sato, N., Burmeister, C., and J. Rey, "Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF)", RFC 4585, DOI 10.17487/RFC4585, July 2006, <http://www.rfc-editor.org/info/rfc4585>.

[RFC4796] Hautakorpi, J. and G. Camarillo, "The Session Description Protocol (SDP) Content Attribute", RFC 4796, DOI 10.17487/RFC4796, February 2007, <http://www.rfc-editor.org/info/rfc4796>.

[RFC5104] Wenger, S., Chandra, U., Westerlund, M., and B. Burman, "Codec Control Messages in the RTP Audio-Visual Profile with Feedback (AVPF)", RFC 5104, DOI 10.17487/RFC5104, February 2008, <http://www.rfc-editor.org/info/rfc5104>.

[RFC5117] Westerlund, M. and S. Wenger, "RTP Topologies", RFC 5117, DOI 10.17487/RFC5117, January 2008, <http://www.rfc-editor.org/info/rfc5117>.

[RFC5124] Ott, J. and E. Carrara, "Extended Secure RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/SAVPF)", RFC 5124, DOI 10.17487/RFC5124, February 2008, <http://www.rfc-editor.org/info/rfc5124>.

[RFC5285] Singer, D. and H. Desineni, "A General Mechanism for RTP Header Extensions", RFC 5285, DOI 10.17487/RFC5285, July 2008, <http://www.rfc-editor.org/info/rfc5285>.

[RFC5576] Lennox, J., Ott, J., and T. Schierl, "Source-Specific Media Attributes in the Session Description Protocol (SDP)", RFC 5576, DOI 10.17487/RFC5576, June 2009, <http://www.rfc-editor.org/info/rfc5576>.

[RFC5764] McGrew, D. and E. Rescorla, "Datagram Transport Layer Security (DTLS) Extension to Establish Keys for the Secure Real-time Transport Protocol (SRTP)", RFC 5764, DOI 10.17487/RFC5764, May 2010, <http://www.rfc-editor.org/info/rfc5764>.

[RFC6236] Johansson, I. and K. Jung, "Negotiation of Generic Image Attributes in the Session Description Protocol (SDP)", RFC 6236, DOI 10.17487/RFC6236, May 2011, <http://www.rfc-editor.org/info/rfc6236>.

[RFC6562] Perkins, C. and JM. Valin, "Guidelines for the Use of Variable Bit Rate Audio with Secure RTP", RFC 6562, DOI 10.17487/RFC6562, March 2012, <http://www.rfc-editor.org/info/rfc6562>.

[RFC7022] Begen, A., Perkins, C., Wing, D., and E. Rescorla, "Guidelines for Choosing RTP Control Protocol (RTCP) Canonical Names (CNAMEs)", RFC 7022, DOI 10.17487/RFC7022, September 2013, <http://www.rfc-editor.org/info/rfc7022>.

[RFC7205] Romanow, A., Botzko, S., Duckworth, M., and R. Even, Ed., "Use Cases for Telepresence Multistreams", RFC 7205, DOI 10.17487/RFC7205, April 2014, <http://www.rfc-editor.org/info/rfc7205>.

Authors’ Addresses

Roni Even
Huawei Technologies
Tel Aviv
Israel

Email: roni.even@mail01.huawei.com

Jonathan Lennox
Vidyo, Inc.
433 Hackensack Avenue
Seventh Floor
Hackensack, NJ 07601
US

Email: jonathan@vidyo.com