The Session Description Protocol (SDP) Application Token Attribute
draft-even-mmusic-application-token-00.txt

Abstract

   The RTP fixed header includes the payload type number and the SSRC
   values of the RTP stream. RTP defines how you de-multiplex streams
   within an RTP session, but in some use cases applications need
   further identifiers in order to identify the application semantics
   associated with particular streams within the session.

   This document defines a mechanism to provide the mapping between the
   SSRCs of RTP streams and the application semantics by defining
   extensions to RTP and RTCP messages.

Status of This Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at http://datatracker.ietf.org/drafts/current/.

   Internet-Drafts are draft documents valid for a maximum of six months
   and may be updated, replaced, or obsoleted by other documents at any
   time. It is inappropriate to use Internet-Drafts as reference
   material or to cite them other than as "work in progress."

   This Internet-Draft will expire on December 30, 2013.

Table of Contents

1. Introduction

   The RTP [RFC3550] header includes the payload type number and the
   SSRC values of the RTP stream. RTP defines how you de-multiplex
   streams within an RTP session, but in some use cases, applications
   need further identifiers in order to identify semantics associated
   with particular streams within the session.

   There is ongoing work to define how to support, using SDP [RFC4566],
   multiple RTP media streams in one or more m-lines that define a
   single RTP session (as specified in [RFC3550]). The work is
   addressing the WebRTC architecture [I-D.ietf-rtcweb-overview], and
   some work will be needed when looking for a general solution in
   MMUSIC that can be used for non-WebRTC systems.

   RTCWEB Plan A [I-D.roach-rtcweb-plan-a] that an m-line in SDP
   represents a single RTP stream. De-multiplexing is done by payload
   type (PT) number (which MUST be unique), and if unique PTs are not
   feasible, use SSRC information in the SDP to identify the RTP stream.

   RTCWEB Plan B [I-D.uberti-rtcweb-plan] takes a different approach,
   and creates a hierarchy within SDP; an m= line defines an "envelope",
   specifying codec and transport parameters, and [RFC5576] a=ssrc lines
   are used to describe individual media sources within that envelope.

Each m-line defines multiple RTP streams. This requires that the
SSRCs of all RTP streams in the session are declared before they
appear as RTP streams.

No plan [I-D.ivov-rtcweb-noplan] proposes using a single m-line for
each media type but does not require that all SSRCs will be declared
in the SDP. The de-multiplexing is done based on the unique PT
numbers and the mapping of SSRC to the application usage may be done
by application protocol. In web application, for example, the
application specific signaling may use something like { "leftSSRC":
"1234", "rightSSRC": "5678" }.

Some applications may require more information about the usage of the
RTP streams. For example, RTP streams from different cameras that
need to be identified by the application in order to render them
correctly, or a source that can send multiple versions of the same
stream in different resolutions (Simulcast
[I-D.westerlund-avtcore-rtp-simulcast]).

SDP provides in [RFC4574] a "label" attribute that contains a token
defined by an application and is used in its context. "Label" can be
attached to m-lines in multiple SDP documents allowing the
application to logically identify the media streams across SDP
sessions when necessary. The "label" attribute is a token and does
not provide any information about the content of the stream.
[RFC4796] defines the "content" attribute providing information about
the content of the stream, currently there is a small set of values
for the content attribute.

Both "label" and "content" attribute are SDP media-level attributes,
so when an SDP m-line supports multiple RTP streams, this value is
applicable to all sources described by the SDP m-line.

There is a need to have a token that will allow the mapping between a
single source (identified by an SSRC) in an m-line to the application
logic (Source may be a single RTP stream identified by a unique
SSRC). For example, SSRC1 is the RTP stream from the left camera and
SSRC2 is the RTP stream from the right camera both can be specified
in a single SDP m-line and may have the same PT number.

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. Proposal for an Application ID token

   As we saw in the previous section, there are tokens defined that
   could be used for the mapping, but they have existing usages and
   semantics, and tend to apply at media-level rather than source-level.
   In order to avoid overload of existing attributes, it is better to
   have a new token attribute that can identify a specific source
   corresponding to the application. This document defines such a new
   token, called "AppID".

   AppID is a general-purpose token associated with an RTP stream,
   allowing the semantics of the stream with a token to be defined by
   the application. This token may be mapped, for example, to a CLUE
   media capture using CLUE protocol [I-D.ietf-clue-framework], or to a
   specific resolution in a simulcast application described in the SDP .

   The token is chosen by the sender, and represents the RTP stream that
   will be sent to the receiver.

   The proposed token can be sent using SDP, RTCP SDES messages
   [RFC3550], or an RTP header extension [RFC5285]

   The SSRC mapping may be available to the receiver when receiving the
   RTP stream through the RTP header extension, but may also be
   available ahead of time via an RTCP SDES message conveyed before the
   source started sending, even if the receiver has not seen any RTP
   packets from this source like in a multipoint conference or in the
   SDP description.

   The receiver can receive new sources that may be of two kinds.

   o A new RTP stream replacing an existing RTP stream, in which case
      the AppID of the replaced RTP stream will be assigned to the new
      SSRC.

   o A new RTP stream requiring a different AppID, for example, when
      adding a presentation stream to an existing call with two video
      cameras from a room.

   The solution should support a RTP session as described using SDP.
   The RTP session may be specified using a single SDP m-line, or using
   Bundle [I-D.ietf-mmusic-sdp-bundle-negotiation], using more than one
   m-line. In the latter case, if the SSRCs of all RTP streams are not
   known in advance, the AppIDs associated with each m-line need to be
   available to the receiver in order to map each SSRC to a specific
   m-line configuration.

   To support these cases the document defines a new SDP media level
   attribute a=appID that can be used to list all the appIDs that an
   application may use.

   The appID syntax provides a token identifier and optional SDP
   attributes that describe the application usage if exists in SDP.
   Application usage in SDP may be, for example, an image attribute
   describing a simulcast application usage
   [I-D.westerlund-avtcore-rtp-simulcast].

   Each value of the AppID maps to one SSRC at a time. When a new SSRC
   is mapped to an existing AppID using an RTP header extension or SDES
   message, it replaces the previous RTP stream for this application
   usage.

   The formal representation of the appID token is:

      appid-attribute = "appID:" tokenlist [SP attribute]

      tokenlist = token *("," token)

      ; The base definition of "attribute" is in [RFC4566].

      ; (It is the content of "a=" lines.)

   Examples:

   The SSRCs of the streams are not known when the SDP offer is sent,
   two appID are specified and can be used for mapping to specific SSRCs
   in the application.

      m=video 49200 RTP/AVP 99

      a=rtpmap:99 H264/90000

      a=appID:2,3

   The second example is when the application usage of the RTP steam is
   specified using SDP to provide different image resolutions.

      m=video 49200 RTP/AVP 98, 99

      a=rtpmap:98 H264/90000

      a=rtpmap:99 H264/90000

      a=appID:2 imageattr:98 send [x=480,y=320] recv *

```
      a=appID:3 imageattr:99 send [x=800,y=640]  recv *
```

3.1. RTCP SDES message

The document specify 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
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |    AppID = XXX    |     length     |AppID token
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |  ....
```

This AppID is the same token as defined in the new SDP attribute and
will also be used in the RTP header extension.

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

3.2. RTP Header Extension

The Application ID could be carried within the RTP header extension
field, using [RFC5285] two bytes header extension.

This is negotiated within the SDP i.e.

a=extmap:1 urn:ietf:params:rtp-hdrext:App-ID

Packets tagged by the sender with the AppID will 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=1           |    Len-1          |     AppID
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |  AppID  ..............     |
   +-+-+-+-+-+-+-+-+
```

To add or modify the AppID by an intermediary can be an expensive
operation, particularly if SRTP is used to authenticate the packet.
Modification to the contents of the RTP header requires a re-
authentication of the complete packet, and this could prove to be a
limiting factor in the throughput of a multipoint device.

There is no need to send the AppID header extension with all RTP
packets. Senders MAY choose to send it only when a new SSRC is sent,
or when an SSRC changes its association to an AppID. 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. For
codecs with decoder refresh points (such as I-Frames in video
codecs), senders also SHOULD send the AppID header extension along
with the packets carrying the decoder refresh.

4. Using Application ID token

The usage of mapping may depend on the de-multiplexing of the RTP
streams in the SDP m-lines. Currently we have three options
discussed based on input from the RTCweb WG.

For plan A [I-D.roach-rtcweb-plan-a], since each RTP stream is
described by a specific m-line it will be enough to have a media
level token for mapping the sent stream.

Only need for example:

    m=video 49200 RTP/AVP 99

    a=rtpmap:99 H264/90000

    a=appID 2

For plan B [I-D.uberti-rtcweb-plan] which adds another level of RTP
stream description, the mapping of SSRC to the application will need
to be at the SSRC level base on [RFC5576] since all SSRCs are
specified in the m-line. The document addresses the mapping of SSRCs
using the SSRC attribute but uses the msid [I-D.ietf-mmusic-msid]
that defines a specific semantics for each SSRC. The following offer
example is using RFC5576 to provide source specific attribute
identifier.

    m=video 49200 RTP/AVP

    a=rtpmap:99 H264/90000

    a=max-send-ssrc:{*:3}

    a=max-recv-ssrc:{*:3}

    a=ssrc:11111 AppID:1

    a=ssrc:22222 AppID:2

        a=ssrc:33333 AppID:3

   When using noplan [I-D.ivov-rtcweb-noplan] in MMUSIC, not all SSRCs
   will be known ahead of time. For example, in the following SDP the
   offer offers either two streams with the same resolution (for example
   two cameras) or two streams with different resolutions.

        m=video 5002 RTP/SAVPF 98

        a=rtpmap:98 H264/90000

        a= appID 1,2 imageattr:98 send [x=800,y=640,sar=1.1,q=0.6] recv *

        a= appID 3,4 imageattr:98 send [x=480,y=320] recv *

        a=max-send-ssrc:{*:2}

   In the CLUE WG case the mapping is from an RTP stream to a CLUE media
   capture specified in the CLUE framework [I-D.ietf-clue-framework].
   The SSRCs of all streams may be known like in PLAN B but there are
   cases where the SDP may not be available so a pre-announce is
   recommended like in the following example.

        m=video 49200 RTP/AVP

        a=rtpmap:99 H264/90000

        a=max-send-ssrc:{*:5}

        a=max-recv-ssrc:{*:3}

        a=ssrc:11111 AppID:1

        a=ssrc:22222 AppID:2

        a=ssrc:33333 AppID:3

        a=appID 4, 5

   The pre-announce is needed since the new RTCP SDES message includes
   only the SSRC and the appID but not the PT. A receiver of the SDES
   message will be able to map the SSRC to a codec configuration based
   on the SDP pre-announced tokens.

5. Acknowledgements

   Place Holder

6. IANA Considerations

   TBD

7. Security Considerations

   TBD.

8. References

8.1. Normative References

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

   [RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V.
              Jacobson, "RTP: A Transport Protocol for Real-Time
              Applications", STD 64, RFC 3550, July 2003.

   [RFC5285] Singer, D. and H. Desineni, "A General Mechanism for RTP
              Header Extensions", RFC 5285, July 2008.

8.2. Informative References

   [I-D.ietf-clue-framework]
              Duckworth, M., Pepperell, A., and S. Wenger, "Framework
              for Telepresence Multi-Streams", draft-ietf-clue-
              framework-10 (work in progress), May 2013.

   [I-D.ietf-mmusic-msid]
              Alvestrand, H., "Cross Session Stream Identification in
              the Session Description Protocol", draft-ietf-mmusic-
              msid-00 (work in progress), February 2013.

   [I-D.ietf-mmusic-sdp-bundle-negotiation]
              Holmberg, C., Alvestrand, H., and C. Jennings,
              "Multiplexing Negotiation Using Session Description
              Protocol (SDP) Port Numbers", draft-ietf-mmusic-sdp-
              bundle-negotiation-04 (work in progress), June 2013.

   [I-D.ietf-rtcweb-overview]
              Alvestrand, H., "Overview: Real Time Protocols for Brower-
              based Applications", draft-ietf-rtcweb-overview-06 (work
              in progress), February 2013.

   [I-D.ivov-rtcweb-noplan]
              Ivov, E., Marocco, E., and P. Thatcher, "No Plan:
              Economical Use of the Offer/Answer Model in WebRTC

                Sessions with Multiple Media Sources", [draft-ivov-rtcweb-noplan-01](#) (work in progress), June 2013.

   [I-D.roach-rtcweb-plan-a]
                Roach, A. and M. Thomson, "Using SDP with Large Numbers of
                Media Flows", [draft-roach-rtcweb-plan-a-00](#) (work in
                progress), May 2013.

   [I-D.uberti-rtcweb-plan]
                Uberti, J., "Plan B: a proposal for signaling multiple
                media sources in WebRTC.", [draft-uberti-rtcweb-plan-00](#)
                (work in progress), May 2013.

   [I-D.westerlund-avtcore-rtp-simulcast]
                Westerlund, M., Lindqvist, M., and F. Jansson, "Using
                Simulcast in RTP Sessions", [draft-westerlund-avtcore-rtp-simulcast-02](#) (work in progress), February 2013.

   [RFC4566]   Handley, M., Jacobson, V., and C. Perkins, "SDP: Session
                Description Protocol", [RFC 4566](#), July 2006.

   [RFC4574]   Levin, O. and G. Camarillo, "The Session Description
                Protocol (SDP) Label Attribute", [RFC 4574](#), August 2006.

   [RFC4575]   Rosenberg, J., Schulzrinne, H., and O. Levin, "A Session
                Initiation Protocol (SIP) Event Package for Conference
                State", [RFC 4575](#), August 2006.

   [RFC4796]   Hautakorpi, J. and G. Camarillo, "The Session Description
                Protocol (SDP) Content Attribute", [RFC 4796](#), February
                2007.

   [RFC5576]   Lennox, J., Ott, J., and T. Schierl, "Source-Specific
                Media Attributes in the Session Description Protocol
                (SDP)", [RFC 5576](#), June 2009.

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

Qin Wu
Huawei Technologies

Email: bill.wu@huawei.com