Network Working Group H. Alvestrand
Internet-Draft Google
Intended status: Standards Track October 15, 2012
Expires: April 18, 2013

Cross Session Stream Identification in the Session Description Protocol
draft-alvestrand-mmusic-msid-01

Abstract

This document specifies a grouping mechanism for RTP media streams that can be used to specify relations between media streams within different RTP sessions.

This mechanism is used to signal the association between the RTP concept of SSRC and the WebRTC concept of "media stream" / "media stream track" using SDP signalling.

This document is an input document for discussion. It should be discussed in the MMUSIC WG list, mmusic@ietf.org.

Requirements Language

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 RFC 2119 [RFC2119].

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

1.1. Structure Of This Document

This document extends the SSRC grouping framework [RFC5888] by adding a new grouping relation that can cross RTP session boundaries.

Section 1.2 gives the background on why a new mechanism is needed.

Section 2 gives the definition of the new mechanism.

Section 4 gives the application of the new mechanism for providing necessary semantic information for the association of MediaStreamTracks to MediaStreams in the WebRTC API .

1.2. Why A New Mechanism Is Needed

When media is carried by RTP [RFC3550], each RTP media stream is distinguished inside an RTP session by its SSRC; each RTP session is distinguished from all other RTP sessions by being on a different transport association (strictly speaking, 2 transport associations, one used for RTP and one used for RTCP, unless RTCP multiplexing [RFC5761] is used).

There exist cases where an application using RTP and SDP needs to signal some relationship between RTP media streams that may be carried in either the same RTP session or different RTP sessions. For instance, there may be a need to signal a relationship between a video track in one RTP session and an audio track in another RTP session. In traditional SDP, it is not possible to signal that these two tracks should be carried in one session, so they are carried in different RTP sessions.

The SSRC grouping mechanism ("a=ssrc-group") [RFC5576] can be used to associate RTP media streams when those RTP media streams are part of the same RTP session. The semantics of this mechanism prevent the association of RTP media streams that are spread across different RTP sessions.

The SDP grouping framework [RFC5888] can be used to group RTP sessions. When an RTP session carries one and only one RTP media stream, it is possible to associate RTP media streams across different RTP sessions. However, if an RTP session has multiple RTP media streams, using multiple SSRCs, the SDP grouping framework cannot be used for this purpose.

There are use cases (some of which are discussed in [I-D.westerlund-avtcore-multiplex-architecture]) where neither of

these approaches is appropriate; In those cases, a new mechanism is needed.

In addition, there is sometimes the need for an application to specify some application-level information about the association between the SSRC and the group. This is not possible using either of the frameworks above.

1.3. Application to the WEBRTC MediaStream

The W3C WebRTC API specification [W3C.WD-webrtc-20120209] specifies that communication between WebRTC entities is done via MediaStreams, which contain MediaStreamTracks. A MediaStreamTrack is generally carried using a single SSRC in an RTP session (forming an RTP media stream. The collision of terminology is unfortunate.) There might possibly be additional SSRCs, possibly within additional RTP sessions, in order to support functionality like forward error correction or simulcast. This complication is ignored below.

In the RTP specification, media streams are identified using the SSRC field. Streams are grouped into RTP Sessions, and also carry a CNAME. Neither CNAME nor RTP session correspond to a MediaStream. Therefore, the association of an RTP media stream to MediaStreams need to be explicitly signalled.

The marking needs to be on a per-SSRC basis, since one RTP session can carry media from multiple MediaStreams, and one MediaStream can have media in multiple RTP sessions. This means that the [RFC4574] "label" attribute, which is used to label RTP sessions, is not usable for this purpose.

The marking needs to also carry the unique identifier of the RTP media stream as a MediaStreamTrack within the media stream; this is done using a single letter to identify whether it belongs in the video or audio track list, and the MediaStreamTrack's position within that array.

This usage is described in Section 4.

2. The Msid Mechanism

This document extends the Source-Specific Media Attributes framework [RFC5576] by adding a new "msid" attribute that can be used with the "a=ssrc" SDP attribute. This new attribute allows endpoints to associate RTP media streams that are carried in different RTP sessions, as well as allowing application-specific information to the association.

The value of the "msid" attribute consists of an identifier and optional application-specific data, according to the following ABNF [RFC5234] grammar:

```
; "attribute" is defined in RFC 4566.
; This attribute should be used with the ssrc-attr from RFC 5576.
attribute =/ msid-attr
msid-attr = "msid:" identifier [ " " appdata ]
identifier = 1*64 ("0".."9" / "a".."z" / "-")
appdata = 1*64 ("0".."9" / "a".."z" / "-")
```

An example MSID value for the SSRC 1234 might look like this:

```
a=ssrc:1234 msid:examplefoo v1
```

The identifier is a string of ASCII characters chosen from 0-9, a-z, A-Z and - (hyphen), consisting of between 1 and 64 characters. It MUST be unique among the identifier values used in the same SDP session. It is RECOMMENDED that is generated using a random-number generator.

Application data is carried on the same line as the identifier, separated from the identifier by a space.

The identifier uniquely identifies a group within the scope of an SDP description.

There may be multiple msid attributes on a single SSRC. There may also be multiple SSRCs that have the same value for identifier and application data.

Endpoints can update the associations between SSRCs as expressed by msid attributes at any time; the semantics and restrictions of such grouping and ungrouping are application dependent.

3. The Msid-Semantic Attribute

In order to fully reproduce the semantics of the SDP and SSRC grouping frameworks, a session-level attribute is defined for signalling the semantics associated with an msid grouping.

This OPTIONAL attribute gives the group identifier and its group semantic; it carries the same meaning as the ssrc-group-attr of RFC 5576 section 4.2, but uses the identifier of the group rather than a list of SSRC values.

The ABNF of msid-semantic is:

```
attribute =/ msid-semantic-attr
msid-semantic-attr = "msid-semantic:" " " identifier token
token = <as defined in RFC 4566>
```

The semantic field may hold values from the IANA registries "Semantics for the "ssrc-group" SDP Attribute" and "Semantics for the "group" SDP Attribute".

An example msid-semantic might look like this:

```
a=msid-semantic: examplefoo LS
```

4. Applying Msid to WebRTC Media Streams

This section creates a new semantic for use with the framework defined in Section 2, to be used for associating SSRCs representing media stream tracks with media streams as defined in [W3C.WD-webrtc-20120209].

The semantic token for WebRTC Media Streams is "WMS".

The value of the msid corresponds to the "id" attribute of a MediaStream. (note: as of Jan 11, 2012, this is called "label". The word "label" means many other things, so the same word should not be used.)

In a WebRTC-compatible SDP description, all SSRCs intending to be sent from one peer will be identified in the SDP generated by that entity.

The appdata for a WebRTC MediaStreamTrack consists of the track type and the track number; the track type is encoded as the single letter "a" (audio) or "v" (video), and the track number is encoded as a decimal integer with no leading zeros. The first track is track zero, and is identified as "a0" for audio, and "v0" for video.

If two different SSRCs have the same value for identifier and appdata, it means that these two SSRCs are both intended for the same MediaStreamTrack. This may occur if the sender wishes to use simulcast or forward error correction, or if the sender intends to switch between multiple codecs on the same MediaStreamTrack.

When an SDP description is updated, a specific msid continues to refer to the same media stream; an msid value MUST NOT be reused for another media stream within a PeerConnection's lifetime.

The following are the rules for handling updates of the list of SSRCs and their msid values.

- o When a new msid value occurs in the description, the recipient can signal to its application that a new media stream has been added.

- o When a description is updated to have more SSRCs with the same msid value, the recipient can signal to its application that new media stream tracks have been added to the media stream.

- o When a description is updated to no longer list the msid value on a specific ssrc, the recipient can signal to its application that the corresponding media stream track has been closed.

- o When a description is updated to no longer list the msid value on any ssrc, the recipient can signal to its application that the media stream has been closed.

OPEN ISSUE: Exactly when should the recipient signal that the track is closed? When the msid value disappears from the description, when the SSRC disappears by the rules of [RFC3550] section 6.3.4 (BYE packet received) and 6.3.5 (timeout), any of the above, or some combination of the above?

4.1. Handling of non-signalled tracks

Pre-WebRTC entities will not send msid. This means that there will be some incoming RTP packets with SSRCs where the recipient does not know about a corresponding MediaStream id.

Handling will depend on whether or not any SSRCs are signalled in the relevant RTP session. There are two cases:

- o No SSRC is signalled with an msid attribute. The SDP session is assumed to be a backwards-compatible session. All incoming SSRCs, on all RTP sessions that are part of the SDP session, are assumed to belong to a single media stream. The identifier of this media stream is "default".

- o Some SSRCs are signalled with an msid attribute. In this case, the session is WebRTC compatible, and the newly arrived SSRCs are either caused by a bug or by timing skew between the arrival of the media packets and the SDP description. These packets MAY be discarded, or they MAY be buffered for a while in order to allow immediate startup of the media stream when the SDP description is updated. The arrival of media packets MUST NOT cause a new MediaStreamTrack to be signalled.

Note: This means that it is wise to include at least one a=ssrc: line with an msid attribute, even when no media streams are yet attached to the session. (Alternative: Mark the RTP session explicitly as "I will signal the media stream tracks explicitly").

It follows from the above that media stream tracks in the "default" media stream cannot be closed by signalling; the application must instead signal these as closed when the SSRC disappears according to the rules of RFC 3550 section 6.3.4 and 6.3.5.

5. IANA Considerations

This document requests IANA to register the "msid" attribute in the "att-field (source level)" registry within the SDP parameters registry, according to the procedures of [RFC5576]

The required information is:

- Contact name, email: IETF, contacted via rtcweb@ietf.org, or a successor address designated by IESG
- Attribute name: msid
- Long-form attribute name: Media stream group Identifier
- The attribute value contains only ASCII characters, and is therefore not subject to the charset attribute.
- The attribute gives an association over a set of SSRCs, potentially in different RTP sessions. It can be used to signal the relationship between a WebRTC MediaStream and a set of SSRCs.
- The details of appropriate values are given in RFC XXXX.

This document requests IANA to register the "WMS" semantic within the "Semantics for the "ssrc-group" SDP Attribute" registry within the SDP parameters registry.

The required information is:

- Description: WebRTC Media Stream, as given in RFC XXXX.
- Token: WMS
- Standards track reference: RFC XXXX

IANA is requested to replace "RFC XXXX" with the RFC number of this

document upon publication.

6. Security Considerations

An adversary with the ability to modify SDP descriptions has the ability to switch around tracks between media streams. This is a special case of the general security consideration that modification of SDP descriptions needs to be confined to entities trusted by the application.

No attacks that are relevant to the browser's security have been identified that depend on this mechanism.

7. Acknowledgements

This note is based on sketches from, among others, Justin Uberti and Cullen Jennings.

Special thanks to Miguel Garcia for his work in reviewing this draft, with many specific language suggestions.

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.

[RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.

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

[W3C.WD-webrtc-20120209]
Bergkvist, A., Burnett, D., Narayanan, A., and C. Jennings, "WebRTC 1.0: Real-time Communication Between Browsers", World Wide Web Consortium WD WD-webrtc-20120209, February 2012, <http://www.w3.org/TR/2012/WD-webrtc-20120209>.

8.2. Informative References

[I-D.westerlund-avtcore-multiplex-architecture]
Westerlund, M., Burman, B., Perkins, C., and H. Alvestrand, "Guidelines for using the Multiplexing Features of RTP", draft-westerlund-avtcore-multiplex-architecture-02 (work in progress), July 2012.

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

[RFC5761] Perkins, C. and M. Westerlund, "Multiplexing RTP Data and Control Packets on a Single Port", RFC 5761, April 2010.

[RFC5888] Camarillo, G. and H. Schulzrinne, "The Session Description Protocol (SDP) Grouping Framework", RFC 5888, June 2010.

Appendix A. Design considerations, open questions and and alternatives

This appendix should be deleted before publication as an RFC.

One suggested mechanism has been to use CNAME instead of a new attribute. This was abandoned because CNAME identifies a synchronization context; one can imagine both wanting to have tracks from the same synchronization context in multiple media streams and wanting to have tracks from multiple synchronization contexts within one media stream.

Another suggestion has been to put the msid value within an attribute of RTCP SR (sender report) packets. This doesn't offer the ability to know that you have seen all the tracks currently configured for a media stream.

There has been a suggestion that this mechanism could be used to mute tracks too. This is not done at the moment.

The special value "default" and the reservation of "example*" seems bothersome; apart from that, it's a random string. It's uncertain whether "example" has any benefit.

An alternative to the "default" media stream is to let each new media stream track without a msid attribute create its own media stream. Input on this question is sought.

Discarding of incoming data when the SDP description isn't updated yet (section 3) may cause clipping. However, the same issue exists

when crypto keys aren't available. Input sought.

There's been a suggestion that acceptable SSRCs should be signalled in a response, giving a recipient the ability to say "no" to certain SSRCs. This is not supported in the current version of this document.

This specification reuses the ssrc-group semantics registry for this semantic, on the argument that the WMS purpose is more similar to an SSRC grouping than a session-level grouping, and allows values from both registries, on the argument that some semantics (like LS) are well defined for MSID. Input sought.

Appendix B. Change log

This appendix should be deleted before publication as an RFC.

B.1. Changes from rtcweb-msid-00 to -01

Added track identifier.

Added inclusion-by-reference of draft-lennox-mmusic-source-selection for track muting.

Some rewording.

B.2. Changes from rtcweb-msid-01 to -02

Split document into sections describing a generic grouping mechanism and sections describing the application of this grouping mechanism to the WebRTC MediaStream concept.

Removed the mechanism for muting tracks, since this is not central to the MSID mechanism.

B.3. Changes from rtcweb-msid-02 to mmusic-msid-00

Changed the draft name according to the wishes of the MMUSIC group chairs.

Added text indicting cases where it's appropriate to have the same appdata for multiple SSRCs.

Minor textual updates.

B.4. Changes from mmusic-msid-00 to -01

Increased the amount of explanatory text, much based on a review by Miguel Garcia.

Removed references to BUNDLE, since that spec is under active discussion.

Removed distinguished values of the MSID identifier.

Author's Address

Harald Alvestrand
Google
Kungsbron 2
Stockholm, 11122
Sweden

Email: harald@alvestrand.no