

Network Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: April 9, 2017

A. Roach  
Mozilla  
S. Nandakumar  
Cisco Systems  
P. Thatcher  
Google  
October 06, 2016

**RTP Stream Identifier Source Description (SDS)**  
**draft-ietf-avtext-rid-09**

Abstract

This document defines and registers two new RTP Stream Identifier Source Description (SDS) items. One, named RtpStreamId, is used for unique identification of RTP streams. The other, RepairedRtpStreamId, can be used to identify which stream a redundancy RTP stream is to be used to repair.

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 April 9, 2017.

Copyright Notice

Copyright (c) 2016 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must

include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

- [1. Introduction](#) . . . . . [2](#)
- [2. Terminology](#) . . . . . [3](#)
- 3. Usage of RtpStreamId and RepairedRtpStreamId in RTP and RTCP [3](#)
  - [3.1. RTCP 'RtpStreamId' SDES Extension](#) . . . . . [5](#)
  - [3.2. RTCP 'RepairedRtpStreamId' SDES Extension](#) . . . . . [5](#)
  - 3.3. RTP 'RtpStreamId' and 'RepairedRtpStreamId' Header Extensions . . . . . [5](#)
- 4. IANA Considerations . . . . . [6](#)
  - [4.1. New RtpStreamId SDES item](#) . . . . . [6](#)
  - [4.2. New RepairRtpStreamId SDES item](#) . . . . . [6](#)
  - [4.3. New RtpStreamId Header Extension URI](#) . . . . . [7](#)
  - [4.4. New RepairRtpStreamId Header Extension URI](#) . . . . . [7](#)
- 5. Security Considerations . . . . . [7](#)
- 6. Acknowledgements . . . . . [8](#)
- 7. References . . . . . [8](#)
  - [7.1. Normative References](#) . . . . . [8](#)
  - [7.2. Informative References](#) . . . . . [9](#)
- Authors' Addresses . . . . . [9](#)

**1. Introduction**

RTP sessions frequently consist of multiple streams, each of which is identified at any given time by its SSRC; however, the SSRC associated with a stream is not guaranteed to be stable over its lifetime. Within a session, these streams can be tagged with a number of identifiers, including CNAMEs and MSIDs [[I-D.ietf-mmusic-msid](#)]. Unfortunately, none of these have the proper ordinality to refer to an individual stream; all such identifiers can appear in more than one stream at a time. While approaches that use unique Payload Types (PTs) per stream have been used in some applications, this is a semantic overloading of that field, and one for which its size is inadequate: in moderately complex systems that use PT to uniquely identify every potential combination of codec configuration and unique stream, it is possible to simply run out of values.

To address this situation, we define a new RTCP Stream Identifier Source Description (SDES) identifier, RtpStreamId, that uniquely identifies a single RTP stream. A key motivator for defining this identifier is the ability to differentiate among different encodings of a single Source Stream that are sent simultaneously (i.e., simulcast). This need for unique identification extends to dependent



streams (e.g., where layers used by a layered codec are transmitted on separate streams).

At the same time, when redundancy RTP streams are in use, we also need an identifier that connects such streams to the RTP stream for which they are providing redundancy. For this purpose, we define an additional SDES identifier, `RepairedRtpStreamId`. This identifier can appear only in packets associated with a redundancy RTP stream. They carry the same value as the `RtpStreamId` of the RTP stream that the redundant RTP stream is correcting.

## 2. Terminology

In this document, the terms "source stream", "RTP stream", "source RTP stream", "dependent stream", "received RTP stream", and "redundancy RTP stream" are used as defined in [\[RFC7656\]](#).

The following acronyms are also used:

- o CNAME: Canonical End-Point Identifier, defined in [\[RFC3550\]](#)
- o MID: Media Identification, defined in [\[I-D.ietf-mmusic-sdp-bundle-negotiation\]](#)
- o MSID: Media Stream Identifier, defined in [\[I-D.ietf-mmusic-msid\]](#)
- o RTCP: Real-time Transport Control Protocol, defined in [\[RFC3550\]](#)
- o RTP: Real-time Transport Protocol, defined in [\[RFC3550\]](#)
- o SDES: Source Description, defined in [\[RFC3550\]](#)
- o SSRC: Synchronization Source, defined in [\[RFC3550\]](#)

## 3. Usage of `RtpStreamId` and `RepairedRtpStreamId` in RTP and RTCP

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; however, in some use cases, applications need further identifiers in order to effectively map the individual RTP Streams to their equivalent payload configurations in the SDP.

This specification defines two new RTCP SDES items [\[RFC3550\]](#). The first item is '`RtpStreamId`', which is used to carry RTP stream identifiers within RTCP SDES packets. This makes it possible for a receiver to associate received RTP packets (identifying the RTP stream) with a media description having the format constraint specified. The second is '`RepairedRtpStreamId`', which can be used in



redundancy RTP streams to indicate the RTP stream repaired by a redundancy RTP stream.

To be clear: the value carried in a RepairedRtpStreamId will always match the RtpStreamId value from another RTP stream in the same session. For example, if a source RTP stream is identified by RtpStreamId "A", then any redundancy RTP stream that repairs that source RTP stream will contain a RepairedRtpStreamId of "A" (if this mechanism is being used to perform such correlation). These redundant RTP streams may also contain their own unique RtpStreamId.

This specification also uses the RTP header extension for RTCP SDES items [[I-D.ietf-avtext-sdes-hdr-ext](#)] to allow carrying RtpStreamId and RepairedRtpStreamId values in RTP packets. This allows correlation at stream startup, or after stream changes where the use of RTCP may not be sufficiently responsive. This speed of response is necessary since, in many cases, the stream cannot be properly processed until it can be identified.

RtpStreamId and RepairedRtpStreamId values are scoped by source identifier (e.g., CNAME) and by media session. When the media is multiplexed using the BUNDLE extension [[I-D.ietf-mmusic-sdp-bundle-negotiation](#)], these values are further scoped by their associated MID values. For example: an RtpStreamId of "1" may be present in the stream identified with a CNAME of "1234@example.com", and may also be present in a stream with a CNAME of "5678@example.org", and these would refer to different streams. Similarly, an RtpStreamId of "1" may be present with an MID of "A", and again with a MID of "B", and also refer to two different streams.

Note that the RepairedRtpStreamId mechanism is limited to indicating one repaired stream per redundancy stream. If systems require correlation for schemes in which a redundancy stream contains information used to repair more than one stream, they will have to use a more complex mechanism than the one defined in this specification.

As with all SDES items, RtpStreamId and RepairedRtpStreamId are limited to a total of 255 octets in length. RtpStreamId and RepairedStreamId are constrained to contain only alphanumeric characters. For avoidance of doubt, the only allowed byte values for these IDs are decimal 48 through 57, 65 through 90, and 97 through 122.



**3.1. RTCP 'RtpStreamId' SDDES Extension**

```

 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|RtpStreamId=TBD|   length   | RtpStreamId           ...
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

```

The RtpStreamId payload is ASCII encoded and is not null-terminated.

RFC EDITOR NOTE: Please replace TBD with the assigned SDDES identifier value.

**3.2. RTCP 'RepairedRtpStreamId' SDDES Extension**

```

 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|Repaired...=TBD|   length   | RepairRtpStreamId     ...
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

```

The RepairedRtpStreamId payload is ASCII encoded and is not null-terminated.

RFC EDITOR NOTE: Please replace TBD with the assigned SDDES identifier value.

**3.3. RTP 'RtpStreamId' and 'RepairedRtpStreamId' Header Extensions**

Because recipients of RTP packets will typically need to know which streams they correspond to immediately upon receipt, this specification also defines a means of carrying RtpStreamId and RepairedRtpStreamId identifiers in RTP extension headers, using the technique described in [[I-D.ietf-avtext-sdes-hdr-ext](#)].

As described in that document, the header extension element can be encoded using either the one-byte or two-byte header, and the identification-tag payload is ASCII-encoded.

As the identifier is included in an RTP header extension, there should be some consideration given to the packet expansion caused by the identifier. To avoid Maximum Transmission Unit (MTU) issues for the RTP packets, the header extension's size needs to be taken into account when encoding media. Note that the set of header extensions included in the packet needs to be padded to the next 32-bit boundary [[RFC5285](#)].



In many cases, a one-byte identifier will be sufficient to distinguish streams in a session; implementations are strongly encouraged to use the shortest identifier that fits their purposes. Implementors are warned, in particular, not to include any information in the identifier that is derived from potentially user-identifying information, such as user ID or IP address. To avoid identification of specific implementations based on their pattern of tag generation, implementations are encouraged to use a simple scheme that starts with the ASCII digit "1", and increments by one for each subsequent identifier.

#### **4. IANA Considerations**

##### **4.1. New RtpStreamId SDES item**

RFC EDITOR NOTE: Please replace RFCXXXX with the RFC number of this document.

RFC EDITOR NOTE: Please replace TBD with the assigned SDES identifier value.

This document adds the RtpStreamId SDES item to the IANA "RTP SDES item types" registry as follows:

Value:	TBD
Abbrev.:	RtpStreamId
Name:	RTP Stream Identifier
Reference:	RFCXXXX

##### **4.2. New RepairRtpStreamId SDES item**

RFC EDITOR NOTE: Please replace RFCXXXX with the RFC number of this document.

RFC EDITOR NOTE: Please replace TBD with the assigned SDES identifier value.

This document adds the RepairedRtpStreamId SDES item to the IANA "RTP SDES item types" registry as follows:

Value:	TBD
Abbrev.:	RepairedRtpStreamId
Name:	Repaired RTP Stream Identifier
Reference:	RFCXXXX



### **[4.3.](#) New RtpStreamId Header Extension URI**

RFC EDITOR NOTE: Please replace RFCXXXX with the RFC number of this document.

This document defines a new extension URI in the RTP SDES Compact Header Extensions sub-registry of the RTP Compact Header Extensions registry sub-registry, as follows

Extension URI: urn:ietf:params:rtp-hdext:sdes:rtp-stream-id  
Description: RTP Stream Identifier Contact: adam@nostrum.com  
Reference: RFCXXXX

### **[4.4.](#) New RepairRtpStreamId Header Extension URI**

RFC EDITOR NOTE: Please replace RFCXXXX with the RFC number of this document.

This document defines a new extension URI in the RTP SDES Compact Header Extensions sub-registry of the RTP Compact Header Extensions registry sub-registry, as follows

Extension URI: urn:ietf:params:rtp-hdext:sdes:repaired-rtp-sream-id  
Description: RTP Repaired Stream Identifier Contact: adam@nostrum.com  
Reference: RFCXXXX

## **[5.](#) Security Considerations**

Although the identifiers defined in this document are limited to be strictly alphanumeric, SDES items have the potential to carry any string. As a consequence, there exists a risk that it might carry privacy-sensitive information. Implementations need to take care when generating identifiers so that they do not contain information that can identify the user or allow for long term tracking of the device. Following the generation recommendations in [Section 3.3](#) will result in non-instance-specific labels, with only minor fingerprinting possibilities in the total number of used RtpStreamIds and RepairedRtpStreamIds.

Even if the SDES items are generated to convey as little information as possible, implementors are strongly encouraged to encrypt SDES items - both in RTCP and RTP header extensions - so as to preserve privacy against third parties.

As the SDES items are used for identification of the RTP streams for different application purposes, it is important that the intended values are received. An attacker, either a third party or malicious RTP middlebox, that removes, or changes the values for these SDES



items, can severely impact the application. The impact can include failure to decode or display the media content of the RTP stream. It can also result in incorrectly attributing media content to identifiers of the media source, such as incorrectly identifying the speaker. To prevent this from occurring due to third party attacks, integrity and source authentication is needed.

Options for Securing RTP Sessions [[RFC7201](#)] discusses options for how encryption, integrity and source authentication can be accomplished.

## 6. Acknowledgements

Many thanks for review and input from Cullen Jennings, Magnus Westerlund, Colin Perkins, Jonathan Lennox, and Paul Kyzivat. Magnus Westerlund provided substantially all of the Security Considerations section.

## 7. References

### 7.1. Normative References

[I-D.ietf-avtext-sdes-hdr-ext]

Westerlund, M., Burman, B., Even, R., and M. Zanaty, "RTP Header Extension for RTCP Source Description Items", [draft-ietf-avtext-sdes-hdr-ext-07](#) (work in progress), June 2016.

[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-32](#) (work in progress), August 2016.

[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>>.

[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>>.

[RFC7656] Lennox, J., Gross, K., Nandakumar, S., Salgueiro, G., and B. Burman, Ed., "A Taxonomy of Semantics and Mechanisms for Real-Time Transport Protocol (RTP) Sources", [RFC 7656](#), DOI 10.17487/RFC7656, November 2015, <<http://www.rfc-editor.org/info/rfc7656>>.



## **7.2. Informative References**

- [I-D.ietf-mmusic-msid]  
Alvestrand, H., "WebRTC MediaStream Identification in the Session Description Protocol", [draft-ietf-mmusic-msid-15](#) (work in progress), July 2016.
- [RFC7201] Westerlund, M. and C. Perkins, "Options for Securing RTP Sessions", [RFC 7201](#), DOI 10.17487/RFC7201, April 2014, <<http://www.rfc-editor.org/info/rfc7201>>.

### Authors' Addresses

Adam Roach  
Mozilla

Email: [adam@nostrum.com](mailto:adam@nostrum.com)

Suhas Nandakumar  
Cisco Systems

Email: [snandaku@cisco.com](mailto:snandaku@cisco.com)

Peter Thatcher  
Google

Email: [pthatcher@google.com](mailto:pthatcher@google.com)

