Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 4, 2016

P. Thatcher
Google
M. Zanaty
S. Nandakumar
Cisco Systems
B. Burman
Ericsson
A. Roach
B. Campen
Mozilla

February 01, 2016

RTP Payload Format Constraints
draft-ietf-mmusic-rid-01

Abstract

In this specification, we define a framework for identifying RTP Streams with the constraints on its payload format in the Session Description Protocol. This framework defines a new "rid" SDP attribute to: a) effectively identify the RID RTP Streams within a RTP Session, b) constrain their payload format parameters in a codec-agnostic way beyond what is provided with the regular Payload Types and c) enable unambiguous mapping between the RID RTP Streams to their media format specification in the SDP.

Status of This Memo

This Internet-Draft will expire on August 4, 2016.

Copyright Notice

Table of Contents

1. Introduction

Payload Type (PT) in RTP provides a mapping between the format of the RTP payload and the media format description specified in the signaling. For applications that use SDP for signaling, the constructs rtpmap and/or fmtp describe the characteristics of the media that is carried in the RTP payload, mapped to a given PT.

Recent advances in standards have given rise to rich multimedia applications requiring support for multiple RTP Streams within a RTP session [I-D.ietf-mmusic-sdp-bundle-negotiation], [I-D.ietf-mmusic-sdp-simulcast] or having to support a large number of codecs. These demands have unearthed challenges inherent with:

- The restricted RTP PT space in specifying the various payload configurations,

- The codec-specific constructs for the payload formats in SDP,

- Missing or underspecified payload format parameters,

- Overloading of PTs to indicate not just codec configurations, but individual streams within an RTP session.

To expand on these points: [RFC3550] assigns 7 bits for the PT in the RTP header. However, the assignment of static mapping of payload codes to payload formats and multiplexing of RTP with other protocols (such as RTCP) could result in limited number of payload type numbers available for the application usage. In scenarios where the number of possible RTP payload configurations exceed the available PT space within a RTP Session, there is need a way to represent the additional constraints on payload configurations and to effectively map a RID RTP Stream to its corresponding constraints. This issue is exacerbated by the increase in techniques such as simulcast and layered codecs, which introduce additional streams into RTP Sessions.

This specification defines a new SDP framework for constraining Source RTP Streams (Section 2.1.10 [I-D.ietf-avtext-rtp-grouping-taxonomy]), along with the SDP attributes to constrain payload formats in a codec-agnostic way. This framework can be thought of as complementary extension to the way the media format parameters are specified in SDP today, via the "a=fmtp" attribute.

This specification makes use of the RTP Stream Identifier SDES RTCP item defined in [I-D.roach-avtext-rid] to provide correlation between the RTP Packets and their format specification in the SDP.

The additional constraints on individual streams are indicated with a new "a=rid" SDP attribute. Note that the parameters communicated via this attribute only serve to further constrain the parameters that are established on a PT format. They do not relax any existing constraints.

As described in Section 6.2.1, this mechanism achieves backwards compatibility via the normal SDP processing rules, which require unknown a= parameters to be ignored. This means that implementations need to be prepared to handle successful offers and answers from other implementations that neither indicate nor honor the constraints requested by this mechanism.

Further, as described in Section 6 and its subsections, this mechanism achieves extensibility by: (a) having offerers include all supported constraints in their offer, abd (b) having answerers ignore "a=rid" lines that specify unknown constraints.

2. Key Words for Requirements

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]

3. Terminology

The terms "Source RTP Stream", "Endpoint", "RTP Session", and "RTP Stream" are used as defined in [I-D.ietf-avtext-rtp-grouping-taxonomy].

The term "RID RTP Stream" is used as defined in [I-D.roach-avtext-rid].

[RFC4566] and [RFC3264] terminology is also used where appropriate.

4. SDP "a=rid" Media Level Attribute

This section defines new SDP media-level attribute [RFC4566], "a=rid". Roughly speaking, this attribute takes the following form (see Section 9 for a formal definition).

```
a=rid:<rid-identifier> <direction> [pt=<fmt-list>;]<constraint>=<value>...
```

An "a=rid" SDP media attribute specifies constraints defining a unique RTP payload configuration identified via the "rid-identifier". This value binds the restriction to the RID RTP Stream identified by its RID SDES item [I-D.roach-avtext-rid].

The "direction" parameter identifies the directionality of the RID RTP Stream; it may be either "send" or "recv".

The optional "pt" parameter lists one or more PT values that can be used in the associated RID RTP Stream. If the parameter is absent, then any of the PT values specified in the corresponding "m=" line may be used.

The list of zero or more codec-agnostic "constraint" parameters (Section 5) describe the restrictions that the corresponding RID RTP Stream will conform to.

This framework MAY be used in combination with the "a=fmtp" SDP attribute for describing the media format parameters for a given RTP Payload Type. In such scenarios, the "a=rid" constraints (Section 5) further constrain the equivalent "a=fmtp" attributes.

A given SDP media description MAY have zero or more "a=rid" lines describing various possible RTP payload configurations. A given "rid-identifier" MUST NOT be repeated in a given media description ("m=" section).

The "a=rid" media attribute MAY be used for any RTP-based media transport. It is not defined for other transports, although other documents may extend its semantics for such transports.

Though the parameters specified by the "rid" constraints follow a syntax similar to session-level and media-level attributes, they are defined independently. All "rid" parameters MUST be registered with IANA, using the registry defined in Section 12

Section 9 gives a formal Augmented Backus-Naur Form (ABNF) [RFC5234] grammar for the "rid" attribute. The "a=rid" media attribute is not dependent on charset.

5. "a=rid" constraints

This section defines the "a=rid" constraints that can be used to restrict the RTP payload encoding format in a codec-agnostic way.

The following constraints are intended to apply to video codecs in a codec-independent fashion.

- max-width, for spatial resolution in pixels. In the case that stream orientation signaling is used to modify the intended display orientation, this attribute refers to the width of the stream when a rotation of zero degrees is encoded.

- max-height, for spatial resolution in pixels. In the case that stream orientation signaling is used to modify the intended display orientation, this attribute refers to the width of the stream when a rotation of zero degrees is encoded.

- max-fps, for frame rate in frames per second. For encoders that do not use a fixed framerate for encoding, this value should constrain the minimum amount of time between frames: the time between any two consecutive frames SHOULD NOT be less than 1/max-fps seconds.

- max-fs, for frame size in pixels per frame. This is the product of frame width and frame height, in pixels, for rectangular frames.

- max-br, for bit rate in bits per second. The restriction applies to the media payload only, and does not include overhead introduced by other layers (e.g., RTP, UDP, IP, or Ethernet). The exact means of keeping within this limit are left up to the implementation, and instantaneous excursions outside the limit are permissible. For any given one-second sliding window, however, the total number of bits in the payload portion of RTP SHOULD NOT exceed the value specified in "max-br."

- max-pps, for pixel rate in pixels per second. This value SHOULD be handled identically to max-fps, after performing the following conversion: max-fps = max-pps / (width * height). If the stream resolution changes, this value is recalculated. Due to this recalculation, excursions outside the specified maximum are possible during near resolution change boundaries.

- max-bpp, for maximum number of bits per pixel, calculated as an average of all samples of any given coded picture. This is expressed as a floating point value, with an allowed range of 0.0001 to 48.0.

All the constraints are optional and are subject to negotiation based on the SDP Offer/Answer rules described in Section 6.

This list is intended to be an initial set of constraints. Future documents may define additional constraints; see Section 12.2. While this document does not define constraints for audio codecs, there is

no reason such constraints should be precluded from definition and registration by other documents.

Section 9 provides formal Augmented Backus-Naur Form(ABNF) [RFC5234] grammar for each of the "a=rid" attributes defined in this section.

6. SDP Offer/Answer Procedures

This section describes the SDP Offer/Answer [RFC3264] procedures when using this framework.

Note that "rid-identifier" values are only required to be unique within a media section ("m-line"); they do not necessarily need to be unique within an entire RTP session. In traditional usage, each media section is sent on its own unique 5-tuple, which provides an unambiguous scope. Similarly, when using BUNDLE [I-D.ietf-mmusic-sdp-bundle-negotiation], MID values associate RTP streams uniquely to a single media description.

6.1. Generating the Initial SDP Offer

For each media description in the offer, the offerer MAY choose to include one or more "a=rid" lines to specify a configuration profile for the given set of RTP Payload Types.

In order to construct a given "a=rid" line, the offerer must follow these steps:

1. It MUST generate a "rid-identifier" that is unique within a media description

2. It MUST set the direction for the "rid-identifier" to one of "send" or "recv"

3. It MAY include a listing of SDP format tokens (usually corresponding to RTP payload types) allowed to appear in the RID RTP Stream. Any Payload Types chosen MUST be a valid payload type for the media section (that is, it must be listed on the "m=" line). The order of the listed formats is significant; the alternatives are listed from (left) most preferred to (right) least preferred. When using RID, this preference overrides the normal codec preference as expressed by format type ordering on the "m="-line, using regular SDP rules.

4. The Offerer then chooses zero or more "a=rid" constraints (Section 5) to be applied to the rid, and adds them to the "a=rid" line.

5. If the offerer wishes the answerer to have the ability to specify a constraint, but does not wish to set a value itself, it MUST include the name of the constraint in the "a=rid" line, but without any indicated value.

Note: If an "a=fmtp" attribute is also used to provide media-format-specific parameters, then the "a=rid" attributes will further constrain the equivalent "a=fmtp" parameters for the given Payload Type for the specified RID RTP Stream.

If a given codec would require an "a=fmtp" line when used without "a=rid" then the offer MUST include a valid corresponding "a=fmtp" line even when using "a=rid".

6.2. Answerer processing the SDP Offer

For each media description in the offer, and for each "a=rid" attribute in the media description, the receiver of the offer will perform the following steps:

6.2.1. 'rid' unaware Answerer

If the receiver doesn't support the framework proposed in this specification, the entire "a=rid" line is ignored following the standard [RFC3264] Offer/Answer rules.

Section 6.1 requires the offer to include a valid "a=fmtp" line for any codecs that otherwise require it (in other words, the "a=rid" line cannot be used to replace "a=fmtp" configuration). As a result, ignoring the "a=rid" line is always guaranteed to result in a valid session description.

6.2.2. 'rid' aware Answerer

If the answerer supports the "a=rid" attribute, the following steps are executed, in order, for each "a=rid" line in a given media description:

1. Extract the rid-identifier from the "a=rid" line and verify its uniqueness. In the case of a duplicate, the entire "a=rid" line, and all "a=rid" lines with rid-identifiers that duplicate this line, are rejected and MUST NOT be included in the SDP Answer.

2. If the "a=rid" line contains a "pt=" parameter, the list of payload types is verified against the list of valid payload types for the media section (that is, those listed on the "m=" line). Any PT missing from the "m=" line is removed from the "pt=" parameter.

3. The answerer ensures that "a=rid" parameters listed are syntactically well formed. In the case of a syntax error, the "a=rid" line is removed.

4. If the "direction" parameter is "recv", The answerer ensures that "a=rid" parameters are supported. In the case of an unsupported parameter, the "a=rid" line is removed.

5. If the "depend" parameter is included, the answerer MUST make sure that the listed rid-identifiers unambiguously match the rid-identifiers in the SDP offer. Any lines that do not are removed.

6. The answerer verifies that the constraining parameters are consistent with at least one of the codecs to be used with the RID RTP Stream. If the "a=rid" line contains a "pt=" parameter, it contains the list of such codecs; otherwise, the list of such codecs is taken from the associated "m=" line. See Section 8 for more detail. If the "a=rid" parameters are incompatible with the other codec properties for all codecs, then the "a=rid" line is removed.

Note that the answerer does not need to understand every constraint present in a "send" line: if a stream sender constrains the stream in a way that the receiver does not understand, this causes no issues with interoperability.

6.3. Generating the SDP Answer

Having performed verification of the SDP offer as described, the answerer shall perform the following steps to generate the SDP answer.

For each "a=rid" line:

1. The sense of of the "direction" parameter is reversed: "send" is changed to "recv", and "recv" is changed to "send".

2. The answerer MAY choose to modify specific "a=rid" constraint value in the answer SDP. In such a case, the modified value MUST be more constrained than the ones specified in the offer. The answer MUST NOT include any constraints that were not present in the offer.

3. The answerer MUST NOT modify the "rid-identifier" present in the offer.

4. If the "a=rid" line contains a "pt=" parameter, the answerer is allowed to remove one or more media formats from a given "a=rid"

line. If the answerer chooses to remove all the media format tokens from an "a=rid" line, the answerer MUST remove the entire "a=rid" line. If the offer did not contain a "pt=" parameter for a given "a=rid" line, then the answer MUST NOT contain a "pt=" parameter in the corresponding line.

5. In cases where the answerer is unable to support the payload configuration specified in a given "a=rid" line in the offer, the answerer MUST remove the corresponding "a=rid" line. This includes situations in which the answerer does not understand one or more of the constraints in an "a=rid" line with a direction of "recv".

Note: in the case that the answerer uses different PT values to represent a codec than the offerer did, the "a=rid" values in the answer use the PT values that are present in its answer.

6.4. Offerer Processing of the SDP Answer

The offerer shall follow these steps when processing the answer:

1. The offerer matches the "a=rid" line in the answer to the "a=rid" line in the offer using the "rid-identifier". If no matching line can be located in the offer, the "a=rid" line is ignored.

2. If the answer contains any constraints that were not present in the offer, then the offerer SHALL consider the "a=rid" line as rejected.

3. If the constraints have been changed between the offer and the answer, the offerer MUST ensure that the modifications can be supported; if they cannot, the SHALL consider the "a=rid" line as rejected.

4. If the "a=rid" line in the answer contains a "pt=" parameter but the offer did not, the offerer SHALL consider the "a=rid" line as rejected.

5. If the "a=rid" line in the answer contains a "pt=" parameter and the offer did as well, the offerer verifies that the list of payload types is a subset of those sent in the corresponding "a=rid" line in the offer. If not, the offerer SHALL consider the "a=rid" line as rejected.

6. If the "a=rid" line contains a "pt=" parameter, the offerer verifies that the attribute values provided in the "a=rid" attributes are consistent with the corresponding codecs and their other parameters. See Section 8 for more detail. If the "a=rid"

parameters are incompatible with the other codec properties, then the "a=rid" line is removed.

7. The offerer verifies that the constraining parameters are consistent with at least one of the codecs to be used with the RID RTP Stream. If the "a=rid" line contains a "pt=" parameter, it contains the list of such codecs; otherwise, the list of such codecs is taken from the associated "m=" line. See Section 8 for more detail. If the "a=rid" parameters are incompatible with the other codec properties for all codecs, then the "a=rid" line SHALL be considered rejected

Any "a=rid" line present in the offer that was not matched by step 1 above SHALL be considered as rejected.

6.5. Modifying the Session

Offers and answers inside an existing session follow the rules for initial session negotiation. Such an offer MAY propose a change in the number of RIDs in use. To avoid race conditions with media, any RIDs with proposed changes SHOULD use a new ID, rather than re-using one from the previous offer/answer exchange. RIDs without proposed changes SHOULD re-use the ID from the previous exchange.

7. Use with Declarative SDP

Although designed predominantly with session negotiation in mind, the "a=rid" attribute can also be used in declarative SDP situations. When used with declarative SDP, any constraints applied to a RID indicate how the sender intends to constrain the stream they are sending.

In declarative use, the "direction" parameter MUST be set to "send" in all "a=rid" lines.

Recipients of declarative SDP may use the indicated constraints to select an RID RTP Stream to decode, based on their needs and capabilities.

8. Interaction with Other Techniques

Historically, a number of other approaches have been defined that allow constraining media streams via SDP parameters. These include:

- Codec-specific configuration set via format parameters ("a=fmtp"); for example, the H.264 "max-fs" format parameter

- o Size restrictions imposed by image attribute attributes ("a=imgattr") [RFC6236]

When the mechanism described in this document is used in conjunction with these other restricting mechanisms, it is intended to impose additional restrictions beyond those communicated in other techniques.

In an offer, this means that "a=rid" lines, when combined with other restrictions on the media stream, are expected to result in a non-empty union. For example, if image attributes are used to indicate that a PT has a minimum width of 640, then specification of "max-width=320" in an "a=rid" line that is then applied to that PT is nonsensical. According to the rules of Section 6.2.2, this will result in the corresponding "a=rid" line being ignored by the recipient.

In an answer, the "a=rid" lines, when combined with the other restrictions on the media stream, are also expected to result in a non-empty union. If the implementation generating an answer wishes to restrict a property of the stream below that which would be allowed by other parameters (e.g., those specified in "a=fmtp" or "a=imgattr"), its only recourse is to remove the "a=rid" line altogether, as described in Section 6.3. If it instead attempts to constrain the stream beyond what is allowed by other mechanisms, then the offerer will ignore the corresponding "a=rid" line, as described in Section 6.4.

9. Formal Grammar

This section gives a formal Augmented Backus-Naur Form (ABNF) [RFC5234] grammar for each of the new media and "a=rid" attributes defined in this document.

```
rid-syntax        = "a=rid:" rid-identifier SP rid-dir
                    [ rid-pt-param-list / rid-param-list ]

rid-identifier    = 1*(alpha-numeric / "-" / "_")

rid-dir           = "send" / "recv"

rid-pt-param-list = SP rid-fmt-list *(";" rid-param)

rid-param-list    = SP rid-param *(";" rid-param)

rid-fmt-list      = "pt=" fmt *( "," fmt )
                     ; fmt defined in {{RFC4566}}
```

```
   rid-param         = rid-width-param
                       / rid-height-param
                       / rid-fps-param
                       / rid-fs-param
                       / rid-br-param
                       / rid-pps-param
                       / rid-bpp-param
                       / rid-depend-param
                       / rid-param-other

   rid-width-param   = "max-width" [ "=" int-param-val ]

   rid-height-param  = "max-height" [ "=" int-param-val ]

   rid-fps-param     = "max-fps" [ "=" int-param-val ]

   rid-fs-param      = "max-fs" [ "=" int-param-val ]

   rid-br-param      = "max-br" [ "=" int-param-val ]

   rid-pps-param     = "max-pps" [ "=" int-param-val ]

   rid-bpp-param     = "max-bpp" [ "=" float-param-val ]

   rid-depend-param  = "depend=" rid-list

   rid-param-other   = 1*(alpha-numeric / "-") [ "=" param-val ]

   rid-list          = rid-identifier *( "," rid-identifier )

   int-param-val     = 1*DIGIT

   float-param-val   = 1*DIGIT "." 1*DIGIT

   param-val         = *( %x20-58 / %x60-7E )
                       ; Any printable character except semicolon
```

10. SDP Examples

Note: see [I-D.ietf-mmusic-sdp-simulcast] for examples of RID used in simulcast scenarios.

10.1. Many Bundled Streams using Many Codecs

In this scenario, the offerer supports the Opus, G.722, G.711 and DTMF audio codecs, and VP8, VP9, H.264 (CBP/CHP, mode 0/1), H.264-SVC (SCBP/SCHP) and H.265 (MP/M10P) for video. An 8-way video call (to a

mixer) is supported (send 1 and receive 7 video streams) by offering 7 video media sections (1 sendrecv at max resolution and 6 recvonly at smaller resolutions), all bundled on the same port, using 3 different resolutions. The resolutions include:

- 1 receive stream of 720p resolution is offered for the active speaker.

- 2 receive streams of 360p resolution are offered for the prior 2 active speakers.

- 4 receive streams of 180p resolution are offered for others in the call.

Expressing all these codecs and resolutions using 32 dynamic PTs (2 audio + 10x3 video) would exhaust the primary dynamic space (96-127). RIDs are used to avoid PT exhaustion and express the resolution constraints.

NOTE: The SDP given below skips few lines to keep the example short and focused, as indicated by either the "..." or the comments inserted.

The offer for this scenario is shown below.

```
...
m=audio 10000 RTP/SAVPF 96 9 8 0 123
a=rtpmap:96 OPUS/48000
a=rtpmap:9 G722/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:0 PCMU/8000
a=rtpmap:123 telephone-event/8000
a=mid:a1
...
m=video 10000 RTP/SAVPF 98 99 100 101 102 103 104 105 106 107
a=rtpmap:98 VP8/90000
a=fmtp:98 max-fs=3600; max-fr=30
a=rtpmap:99 VP9/90000
a=fmtp:99 max-fs=3600; max-fr=30
a=rtpmap:100 H264/90000
a=fmtp:100 profile-level-id=42401f; packetization-mode=0
a=rtpmap:101 H264/90000
a=fmtp:101 profile-level-id=42401f; packetization-mode=1
a=rtpmap:102 H264/90000
a=fmtp:102 profile-level-id=640c1f; packetization-mode=0
a=rtpmap:103 H264/90000
a=fmtp:103 profile-level-id=640c1f; packetization-mode=1
a=rtpmap:104 H264-SVC/90000
```

```
   a=fmtp:104 profile-level-id=530c1f
   a=rtpmap:105 H264-SVC/90000
   a=fmtp:105 profile-level-id=560c1f
   a=rtpmap:106 H265/90000
   a=fmtp:106 profile-id=1; level-id=93
   a=rtpmap:107 H265/90000
   a=fmtp:107 profile-id=2; level-id=93
   a=sendrecv
   a=mid:v1 (max resolution)
   a=rid:1 send max-width=1280;max-height=720;max-fps=30
   a=rid:2 recv max-width=1280;max-height=720;max-fps=30
   ...
   m=video 10000 RTP/SAVPF 98 99 100 101 102 103 104 105 106 107
   ...same rtpmap/fmtp as above...
   a=recvonly
   a=mid:v2 (medium resolution)
   a=rid:3 recv max-width=640;max-height=360;max-fps=15
   ...
   m=video 10000 RTP/SAVPF 98 99 100 101 102 103 104 105 106 107
   ...same rtpmap/fmtp as above...
   a=recvonly
   a=mid:v3 (medium resolution)
   a=rid:3 recv max-width=640;max-height=360;max-fps=15
   ...
   m=video 10000 RTP/SAVPF 98 99 100 101 102 103 104 105 106 107
   ...same rtpmap/fmtp as above...
   a=recvonly
   a=mid:v4 (small resolution)
   a=rid:4 recv max-width=320;max-height=180;max-fps=15
   ...
   m=video 10000 RTP/SAVPF 98 99 100 101 102 103 104 105 106 107
   ...same rtpmap/fmtp as above...
   ...same rid:4 as above for mid:v5,v6,v7 (small resolution)...
   ...
```

10.2. Scalable Layers

Adding scalable layers to a session within a multiparty conference gives a selective forwarding unit (SFU) further flexibility to selectively forward packets from a source that best match the bandwidth and capabilities of diverse receivers. Scalable encodings have dependencies between layers, unlike independent simulcast streams. RIDs can be used to express these dependencies using the "depend" parameter. In the example below, the highest resolution is offered to be sent as 2 scalable temporal layers (using MRST).

```
Offer:
...
m=audio ...same as previous example ...
...
m=video ...same as previous example ...
...same rtpmap/fmtp as previous example ...
a=sendrecv
a=mid:v1 (max resolution)
a=rid:0 send max-width=1280;max-height=720;max-fps=15
a=rid:1 send max-width=1280;max-height=720;max-fps=30;depend=0
a=rid:2 recv max-width=1280;max-height=720;max-fps=30
a=rid:5 send max-width=640;max-height=360;max-fps=15
a=rid:6 send max-width=320;max-height=180;max-fps=15
a=simulcast: send rid=0;1;5;6 recv rid=2
...
...same m=video sections as previous example for mid:v2-v7...
...
```

11. Open Issues

11.1. Declarative SDP

Section 7 describes the use of "a=rid" for declarative SDP. This is a pretty small amount of work, and the use of this mechanism to describe how a sender is going to constrain a stream does have some amount of utility. Is the text sufficient? If not, do we want to invest the work needed to make RID work with declarative use cases?

PROPOSAL: Keep the current text.

11.2. Definition of bitrate

Some questions have been raised as to whether we need a more formal description of bitrate than we currently use.

If I read correctly, Magnus indicated that the definition in the document is consistent with TIAS, and believes it is sufficiently well defined.

PROPOSAL: keep current definition that exists in description of "max-br".

11.3. Escaping new constraint values

The parameters on an "a=rid:" line are extensible. The syntax for these is:

```
rid-param-other = 1*(alpha-numeric / "-") [ "=" param-val ]

param-val         = *( %x20-58 / %x60-7E )
                    ; Any printable character except semicolon
```

If an extension has values that can contain semicolons, they need an escaping mechanism. Note that this is not an issue for any currently defined parameters, as they all take numeric values only.

1. Change extension syntax to only allow numeric values

2. Define a universal escaping mechanism for all extensions to use

3. Leave this problem for the first extension parameter - if any - to define value in a way that might allow a semicolon. Note that this approach would allow the use of percent-style escaping (e.g., "%3B") but not backslash-style escaping (e.g., "\;"), as parsers that do not support the new constraint would interpret the embedded semicolon as a separator.

PROPOSAL: Option #3

11.4. Utility of max-width and max height

Comment from Stephan Wenger: Are max-width and max-height actually useful controls? Shouldn't max-fs be sufficient for any plausible uses?

PROPOSAL: Keep max-height and max-width. Implementation is well-defined and easily implemented. At least one participant expressed support for these parameters at IETF 94 face-to-face meeting.

11.5. Definition of max-fps

Comment from Stephan Wenger: Would it be better to define max-fps as constraining the average over a second rather than the inverse of the smallest allowed interval between frames?

PROPOSAL: Keep as currently defined. The difference is subtle. The only kinds of cases allowed by an average that aren't allowed by a minimum interframe interval are those such as sending no packets for most of a second, followed by a burst of 30 frames 1 ms apart, as part of a stream constrained to 30 fps. Such cases seem undesirable.

12. IANA Considerations

12.1. New SDP Media-Level attribute

This document defines "rid" as SDP media-level attribute. This attribute must be registered by IANA under "Session Description Protocol (SDP) Parameters" under "att-field (media level only)".

The "rid" attribute is used to identify characteristics of RTP stream with in a RTP Session. Its format is defined in Section 9.

12.2. Registry for RID-Level Parameters

This specification creates a new IANA registry named "att-field (rid level)" within the SDP parameters registry. The "a=rid" parameters MUST be registered with IANA and documented under the same rules as for SDP session-level and media-level attributes as specified in [RFC4566].

Parameters for "a=rid" lines that modify the nature of encoded media MUST be of the form that the result of applying the modification to the stream results in a stream that still complies with the other parameters that affect the media. In other words, parameters always have to restrict the definition to be a subset of what is otherwise allowable, and never expand it.

New parameter registrations are accepted according to the "Specification Required" policy of [RFC5226], provided that the specification includes the following information:

- contact name, email address, and telephone number

- parameter name (as it will appear in SDP)

- long-form parameter name in English

- whether the parameter value is subject to the charset attribute

- an explanation of the purpose of the parameter

- a specification of appropriate attribute values for this parameter

- an ABNF definition of the parameter

The initial set of "a=rid" parameter names, with definitions in Section 5 of this document, is given below:

Type	SDP Name	Reference
att-field	(rid level)	
	max-width	[RFCXXXX]
	max-height	[RFCXXXX]
	max-fps	[RFCXXXX]
	max-fs	[RFCXXXX]
	max-br	[RFCXXXX]
	max-pps	[RFCXXXX]
	max-bpp	[RFCXXXX]
	depend	[RFCXXXX]

It is conceivable that a future document wants to define a RID-level parameter that contains string values. These extensions need to take care to conform to the ABNF defined for rid-param-other. In particular, this means that such extensions will need to define escaping mechanisms if they want to allow semicolons, unprintable characters, or byte values greater than 127 in the string.

13. Security Considerations

As with most SDP parameters, a failure to provide integrity protection over the "a=rid" attributes provides attackers a way to modify the session in potentially unwanted ways. This could result in an implementation sending greater amounts of data than a recipient wishes to receive. In general, however, since the "a=rid" attribute can only restrict a stream to be a subset of what is otherwise allowable, modification of the value cannot result in a stream that is of higher bandwidth than would be sent to an implementation that does not support this mechanism.

The actual identifiers used for RIDs are expected to be opaque. As such, they are not expected to contain information that would be sensitive, were it observed by third-parties.

14. Acknowledgements

Many thanks to review from Cullen Jennings, Magnus Westerlund, and Paul Kyzivat.

15. References

15.1. Normative References

[I-D.roach-avtext-rid]
Roach, A., Nandakumar, S., and P. Thatcher, "RTP Payload Format Constraints", draft-roach-avtext-rid-01 (work in progress), February 2016.

[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>.

[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>.

[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>.

[RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, DOI 10.17487/RFC5234, January 2008, <http://www.rfc-editor.org/info/rfc5234>.

[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>.

15.2. Informative References

[I-D.ietf-avtext-rtp-grouping-taxonomy]
Lennox, J., Gross, K., Nandakumar, S., Salgueiro, G., and B. Burman, "A Taxonomy of Semantics and Mechanisms for Real-Time Transport Protocol (RTP) Sources", draft-ietf-avtext-rtp-grouping-taxonomy-08 (work in progress), July 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-25 (work in progress), January 2016.

[I-D.ietf-mmusic-sdp-simulcast]
Burman, B., Westerlund, M., Nandakumar, S., and M. Zanaty, "Using Simulcast in SDP and RTP Sessions", draft-ietf-mmusic-sdp-simulcast-03 (work in progress), October 2015.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, DOI 10.17487/RFC5226, May 2008, <http://www.rfc-editor.org/info/rfc5226>.

[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>.

Authors' Addresses

Peter Thatcher
Google

Email: pthatcher@google.com

Mo Zanaty
Cisco Systems

Email: mzanaty@cisco.com

Suhas Nandakumar
Cisco Systems

Email: snandaku@cisco.com

Bo Burman
Ericsson

Email: bo.burman@ericsson.com

Adam Roach
Mozilla

Email: adam@nostrum.com

Byron Campen
Mozilla

Email: bcampen@mozilla.com