Network Working Group	H. T. Alvestrand
Internet-Draft	Google
Intended status: Standards Track	September 28, 2011
Expires: March 31, 2012	

SDP Grouping for Single RTP Sessions
draft-alvestrand-one-rtp-02

Abstract

This document describes an extension to the Session Description Protocol (SDP) to describe RTP sessions where media of multiple top level types, for example audio and video, are carried in the same RTP session.

This document is presented to the RTCWEB, AVTCORE and MMUSIC WGs for consideration.

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]*.

Table of Contents

1. Introduction

In the work with the RTCWEB specifications [I-D.ietf-rtcweb-overview], a need was discovered for representing within the SDP framework an SDP session consisting of a single RTP session, where that single RTP session, mapped to a single transport flow, contained multiple top-level data types.
This is advantageous for the use case where there is no desire for different treatment by the network of the different flows in the session, there exist other appropriate mechanisms (for instance based on SSRC) to identify the flows in the session to the applications, and where the handling of multiple RTP sessions would increase the work required to establish the session (for instance by requiring multiple ICE [RFC5245] negotiations, or handling of failure cases where one RTP session is established and another is not).
This document describes how to represent such a session.

2. Requirements for a solution

The requirements for our representation are:

* It should be possible to represent an SDP session consisting of a single RTP session, where that session carries both audio and video.

* If this description is presented in an Offer in the offer/answer model to an entity that does not understand it, the resulting Answer should contain a valid description of an SDP session consisting of one video RTP session and one audio RTP session.

3. SDP Grouping Framework Parameter

This document defines a new semantics extension called TOGETHER within the SDP Grouping framework [RFC5888].
The extension looks like this:
a=group:TOGETHER <first> <subsequent>...
The first media section mentioned in the list is special; it identifies the base media section for the group. This is referred to as the "first" media section below, but it may occur anywhere in the SDP description.
If this semantics extension is present in an SDP Session-level a=group: line, the semantics are that the two or more media sections are intended to be read as components of the description of a single RTP session, creating a single SSRC numbering space that can contain components of all the types described in the referenced media sections.
The following properties of the media sections are REQUIRED:

* The defined RTPMAP values of the section MUST NOT overlap

* The "proto" profile of the sections MUST be the same (e.g RTP/AVPF)

The media sections MAY contain connection data (port numbers or ICE parameters), but some of these may be ignored in processing (see next section).
The reason for the requirement for systematic proto is that there are many combinations that don't make sense (for instance "RTP/AVPF" in one section and "RTP/SAVP" in another would make encryption and availability of TMMBR depend on the outcome of negotiation, which seems strange). The cases where combinations make sense (RTP/AVPF with UDP/FEC for instance) also usually require that separate RTP sessions be used.
It is RECOMMENDED that the source port number of the media sections are the same, since this will give the least difficulty for SDP-parsing intermediaries when trying to keep track of the media flows established as a result of negotiation.

4. Use in Offer/Answer

This extension MAY be included in a Offer; as specified in RFC 5888 section 9 when describing SIP usage, if it is not included in an Offer, it MUST NOT be included in an answer.
If the responder understands the semantics of the TOGETHER extension, the parameters of the first section MUST be used to establish the RTP session, and the parameters for the other sections MUST be ignored.
The following parameters are taken from the first section only:

* Port number from the m= line

* All media-level attributes defined in RFC 5245 section 15.1 - this includes "candidate", "remote-candidates", "ice-mismatch", "ice-ufrag", "ice-pwd"

The bandwidth of the "m" line is treated specially: The values for all "m=" lines in the group are added together, and the resulting value is taken to be the negotiated bandwidth value for the RTP session.
The expected behaviour when the extension is present in an offer and not understood is that the generated answer will not contain the "a=group:TOGETHER" line, and that each sections' parameters will be used.
If the answerer wishes to refuse an entire m= section, while accepting the RTP session otherwise, he MAY indicate this by setting the port number of the relevant section to zero. In all other cases, the port number of second and subsequent sections is to be ignored.
The answerer cannot refuse the first m= section and establish the call.

5. Parameter combining

The general approach taken when combining the sections is to treat the parameters according to normal processing; for instance, a=fmtp:<payloadtype> parameters still bind to their . However, a number of special considerations have to apply.

- *For the "a=rtpmap" lines, their interpretation depends on the "m=" line they occur under, even after combination; "a=rtpmap:8 PCMA/8000" needs to carry the info that it occured under an "m=audio" line.

- *For parameters that can only occur once in a section, such as the port number from the m= line, the occurence of the parameter that comes from the first section needs to take precedence.

- *The "b=" (bandwidth) line needs to be considered specially, since the values are to be added together.

- *Media-level ICE attributes need to come from the first section only, even though syntax would allow more occurences.

6. Interaction with other extensions

If other extensions modify the bandwidth calculation algorithm, those extensions will have to take into consideration how bandwidth from multiple sections of the SDP description should be merged.

7. RTCP bandwidth considerations

A concern has been raised that when audio and video are combined, the bandwidth of RTCP reports required for an audio stream may exceed the bandwidth of the audio stream itself, which seems a bit bizarre. While not critical (overall RTCP bandwidth is still limited to 5% of the total bandwidth), this warrants a little more study.
Considering a combined RTP session with one sender and one recipient, four 1-Mbit/sec video flows and four 100-Kbit/sec audio flows, all flowing in one direction.
The total bandwidth is 4.4 Mbit/sec, so if the RTCP share of the bandwidth is 5% as recommended by [RFC3550] section 6.2, the RTCP bandwidth limit is 220 Kbits/sec. Eight SSRCs need to be reported on. Each report sender will have 24.4 Kbits/second of RTCP bandwidth at its disposal. Assuming a packet size of 100 bytes (11 bytes per SSRC reported on), the maximum RTCP rate allowed is 30 RTCP packets per second, which is slightly slower than the typical audio heartbeat flow of 50 packets per second (20 ms interval).
If this is deemed excessive, one can adopt the RTP/AVPF model of 5-second regular RTCP reports with additional availability of "on-demand" RTCP packets. But the RTCP feedback interval also enters into congestion control algorithms, which may complicate the picture.

8. Examples

The examples are taken from RFC 4317, "SDP Offer/Answer Examples".
Offer

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
a=group:TOGETHER foo bar
m=audio 49170 RTP/AVP 0 8 97
a=mid:foo
b=AS:200
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:97 iLBC/8000
m=video 49170 RTP/AVP 31 32
a=mid:bar
b=AS:1000
a=rtpmap:31 H261/90000
a=rtpmap:32 MPV/90000
```

This is a request to have both audio and video sent over port 49170, and invites the responder to accept these on the same port, creating a single RTP session. If this can't be done, audio and video will be sent over port 49170, but the respondent's different port numbers will be used, creating different 5-tuples for the two RTP sessions. The total bandwidth, if combined, is 1200 Kbits/second; if separated, 200 Kbits goes to audio and 1000 Kbits goes to video.
Answer, from an entity that understands TOGETHER

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
a=group:TOGETHER foo bar
m=audio 49174 RTP/AVP 0
a=mid:foo
b=AS:200
a=rtpmap:0 PCMU/8000
m=video 49174 RTP/AVP 32
a=mid:bar
b=AS:1000
a=rtpmap:32 MPV/90000
```

After processing this answer, video and audio will flow together on one RTP session between initiator port 49170 and responder port 49174.

Answer, from an entity that understands grouping, but does not understand TOGETHER

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49174 RTP/AVP 0
a=mid:foo
a=rtpmap:0 PCMU/8000
b=AS:200
m=video 49175 RTP/AVP 32
a=mid:bar
a=rtpmap:32 MPV/90000
b=AS:1000
```

After processing this answer, video will flow between initiator port 49170 and responder port 49174, while audio flows between initiator port 49170 and responder port 49175, forming two RTP sessions.
Answer, from an entity that does not understand grouping

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49174 RTP/AVP 0
a=rtpmap:0 PCMU/8000
b=AS:200
m=video 49175 RTP/AVP 32
a=rtpmap:32 MPV/90000
b=AS:1000
```

After processing this answer, the flows set up will be the same ones as in the previous example.

9. IANA Considerations

This document requests IANA to register the new SDP Grouping semantic extension called TOGETHER.

10. Security Considerations

No new security issues have been raised specifically for this extension.
Third-party interceptors that sniff negotiation but do not understand the extension may end up listening to the wrong port number for some of the media flows. This is not deemed greatly harmful.

11. Acknowledgements

This draft is based on a discussion between a number of participants at the Quebec City IETF, July 2011, about the issue of multiplexing audio and video on a single network transport using RTP.
Thanks to (in alphabetical order) Christer Holmberg, Randell Jesup, Paul Kyzivat, Muthu Arul Pozhi Perumal, Justin Uberti for reviews and comments.

12. References

12.1. 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.
[RFC5245]	Rosenberg, J., "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols", RFC 5245, April 2010.
[RFC5888]	Camarillo, G. and H. Schulzrinne, "The Session Description Protocol (SDP) Grouping Framework", RFC 5888, June 2010.

12.2. Informative References

[I-D.ietf-rtcweb-overview]	Alvestrand, H, "Overview: Real Time Protocols for Brower-based Applications", Internet-Draft draft-ietf-rtcweb-overview-02, September 2011.
[RFC5761]	Perkins, C. and M. Westerlund, "Multiplexing RTP Data and Control Packets on a Single Port", RFC 5761, April 2010.
[RFC6051]	Perkins, C. and T. Schierl, "Rapid Synchronisation of RTP Flows", RFC 6051, November 2010.

Appendix A. Change log

Appendix A.1. From draft-alvestrand-one-rtp-00 to -01

Added change log and "other matters" appendix.
Added section on parameter combining.
Added an example for an entity that does not understand grouping.
Added port-zero and no-codecs methods for refusing a section.

Appendix A.2. From draft-alvestrand-one-rtp-01 to -02

Clarified that "first" refers to the first section in a TOGETHER section.
Allowed and encouraged the port numbers of sections to be equal (after mmusic discussion).
Added some more text describing the result of processing the example answers.

Appendix B. Other matters

This appendix is not normative.
A number of matters have been raised in discussion on this draft. Some of the discussions have led to actual changes, others have pointed to the need to identify other documents as authoritative on these issues. This appendix serves to collect such matters.

Appendix B.1. RTCP session establishment

This document does not specify anything about RTCP sessions. Since a purpose of the specification is to minimize the number of transport connections, it's natural to use the "a=rtcp-mux" attribute defined in [RFC5761], but this document does not specify anything about that matter.
If the TOGETHER extension is successfully negotiated, there will be only one RTCP session for the RTP session described by the TOGETHER group.
Since the purpose of this extension is to have SSRCs carrying multiple media types on one RTP session, it may be more important than usual to get SSRC metainformation such as CNAME quicly; this can be done by using the recommendation in [RFC6051]to send an RTCP SR at once when starting the flow, and the recipient can use the RTCP-SR-REQ defined in the same document to request an RTCP SR if the initial one is lost.

Appendix B.2. Renegotiation

After the session has been established, there might occur a need to change its parameters. At the moment, no issues that are different from the issues for a session that does not use TOGETHER have been identified.
If a renegotiation mechanism that uses SDP fragments, rather than whole SDP descriptors, is developed, there might be a need for specifying that all the fragments included in the TOGETHER group are sent together. This will have to wait for the development of such a mechanism.

Appendix B.3. ICE negotiation sequence

When an SDP agent sends out an offer using TOGETHER and ICE, it must choose one of two strategies:

* List ICE parameters for all sections. This means that candidate gathering and opening of local ports must happen before the SDP can be generated, and some of the ports get closed, unused, after the answer has been received. Since the generation of candidates may involve negotiation with a remote server such as a TURN server, this may be an expensive proposition. If the same port number is used, the same ICE parameters can be used for all sections too.

* List ICE parameters for only the first section. If the TOGETHER extension is not understood, this will lead to only the first section's RTP session being established. It is up to the application whether or not to consider this to be a call failure.

The choice between these two options is left as a local matter.
The offerer may also choose to send out ICE probes on either one port or all ports even before the answer comes back, in order to speed up connection establishment; this too is left as a local matter.

Appendix B.4. SDP-inspecting intermediaries

There may exist intermediaries that inspect, but do not modify, the SDP flow, and take some action based on what they parse there.
Such intermediaries are architecturally unsound in general (if they really are needed, they should be explicit participants and be able to refuse to carry the extension they don't understand; if not, their working is of no concern to the callers), but they still exist. Some of them will even take active steps to cause a media stream to be closed if they don't see any packets on the port pair that they interpret as having been established by the exchange, causing difficulty for the endpoints. The use of the same port number for all media sections on the offer mitigates this issue.
In the RTCWEB use case, they may have more problems existing, since the negotiation will normally be carried over encrypted HTTPS connections. In other cases, where negotiation is done in the clear, they may be more common.

Author's Address

Harald Tveit Alvestrand Alvestrand Google Kungsbron 2
Stockholm, 11122 Sweden EMail: harald@alvestrand.no