Network Working Group E. Ivov, Ed.
Internet-Draft SIP Communicator
Intended status: Informational E. Marocco, Ed.
Expires: August 22, 2011 Telecom Italia
J. Lennox
Vidyo, Inc.
February 18, 2011

A Real-Time Transport Protocol (RTP) Header Extension for Mixer-to-Client Audio Level Indication
draft-ietf-avtext-mixer-to-client-audio-level-00

Abstract

This document describes a mechanism for RTP-level mixers in audio conferences to deliver information about the audio level of the individual participants. Such audio level indicators are transported in the same RTP packets as the audio data they pertain to.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

The Framework for Conferencing with the Session Initiation Protocol (SIP) defined in RFC 4353 [RFC4353] presents an overall architecture for multi-party conferencing. Among others, the framework borrows from RTP [RFC3550] and extends the concept of a mixer entity "responsible for combining the media streams that make up a conference, and generating one or more output streams that are delivered to recipients". Every participant would hence receive, in a flat single stream, media originating from all the others.

Using such centralized mixer-based architectures simplifies support for conference calls on the client side since they would hardly differ from one-to-one conversations. However, the method also introduces a few limitations. The flat nature of the streams that a mixer would output and send to participants makes it difficult for users to identify the original source of what they are hearing.

Mechanisms that allow the mixer to send to participants cues on current speakers (e.g. the CSRC fields in RTP [RFC3550]) only work for speaking/silent binary indications. There are, however, a number of use cases where one would require more detailed information. Possible examples include the presence of background chat/noise/ music/typing, someone breathing noisily in their microphone, or other cases where identifying the source of the disturbance would make it easy to remove it (e.g. by sending a private IM to the concerned party asking them to mute their microphone). A more advanced scenario could involve an intense discussion between multiple participants that the user does not personally know. Audio level information would help better recognize the speakers by associating with them complex (but still human readable) characteristics like loudness and speed for example.

One way of presenting such information in a user friendly manner would be for a conferencing client to attach audio level indicators to the corresponding participant related components in the user interface as displayed in Figure 1.


```
                        ________________________
                       |                        |
                       |  00:42 |  Weekly Call  |
                       |________________________|
                       |                        |
                       |                        |
                       | Alice |======    | (S) |
                       |                        |
                       | Bob   |=         |     |
                       |                        |
                       | Carol |          | (M) |
                       |                        |
                       | Dave  |===       |     |
                       |                        |
                       |________________________|
```

Figure 1: Displaying detailed speaker information to the user by including audio level for every participant.

Implementing a user interface like the above requires analysis of the media sent from other participants. In a conventional audio conference this is only possible for the mixer since all other conference participants are generally receiving a single, flat audio stream and have therefore no immediate way of determining individual audio levels.

This document specifies an RTP extension header that allows such mixers to deliver audio level information to conference participants by including it directly in the RTP packets transporting the corresponding audio data.

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 RFC 2119 [RFC2119].

3. Protocol Operation

According to RFC 3550 [RFC3550] a mixer is expected to include in outgoing RTP packets a list of identifiers (CSRC IDs) indicating the sources that contributed to the resulting stream. The presence of such CSRC IDs allows an RTP client to determine, in a binary way, the active speaker(s) in any given moment. RTCP also provides a basic mechanism to map the CSRC IDs to user identities through the CNAME

field. More advanced mechanisms, may exist depending on the signaling protocol used to establish and control a conference. In the case of the Session Initiation Protocol [RFC3261] for example, the Event Package for Conference State [RFC4575] defines a <src-id> tag which binds CSRC IDs to media streams and SIP URIs.

This document describes an RTP header extension that allows mixers to indicate the audio-level of every conference participant (CSRC) in addition to simply indicating their on/off status. This new header extension is based on the "General Mechanism for RTP Header Extensions" [RFC5285].

Each instance of this header contains a list of one-octet audio levels expressed in -dBov, with values from 0 to 127 representing 0 to -127 dBov(see Section 4 and Section 5).

Every audio level value pertains to the CSRC identifier located at the corresponding position in the CSRC list. In other words, the first value would indicate the audio level of the conference participant represented by the first CSRC identifier in that packet and so forth. The number and order of these values MUST therefore match the number and order of the CSRC IDs present in the same packet.

When encoding audio level information, a mixer SHOULD include in a packet information that corresponds to the audio data being transported in that same packet. It is important that these values follow the actual stream as closely as possible. Therefore a mixer SHOULD also calculate the values after the original contributing stream has undergone possible processing such as level normalization, and noise reduction for example.

Note that in some cases a mixer may be sending an RTP audio stream that only contains audio level information and no actual audio. Updating a (web) interface conference module may be one reason for this to happen.

It may sometimes happen that a conference involves more than a single mixer. In such cases each of the mixers MAY choose to relay the CSRC list and audio-level information they receive from peer mixers (as long as the total CSRC count remains below 16). Given that the maximum audio level is not precisely defined by this specification, it is likely that in such situations average audio levels would be perceptibly different for the participants located behind the different mixers.

4. Header Format

The audio level indicators are delivered to the receivers in-band using the "General Mechanism for RTP Header Extensions" [RFC5285]. The payload of this extension is an ordered sequence of 8-bit audio level indicators encoded as per Section 5.

```
 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   |  len  |0|  level 1    |0|  level 2    |0|  level 3    ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: Audio level indicators extension format

The 4-bit len field is the number minus one of data bytes (i.e. audio level values) transported in this header extension element following the one-byte header. Therefore, the value zero in this field indicates that one byte of data follows. A value of 15 is not allowed by this specification and it MUST NOT be used as the RTP header can carry a maximum of 15 CSRC IDs. The maximum value allowed is therefore 14 indicating a following sequence of 15 audio level values.

Note that use of the two-byte header defined in RFC 5285 [RFC5285] follows the same rules the only change being the length of the ID and len fields.

5. Audio level encoding

Audio level indicators are encoded in the same manner as audio noise level in the RTP Payload Comfort Noise specification [RFC3389] and audio level in the RTP Extension Header for Client-to-mixer Audio Level Notification [I-D.lennox-avt-rtp-audio-level-exthdr] specification. The magnitude of the audio level is packed into the least significant bits of one audio-level byte with the most significant bit unused and always set to 0 as shown below in Figure 3.


```
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|0|   level     |
+-+-+-+-+-+-+-+-+
```

Figure 3: Audio Level Encoding

The audio level is expressed in -dBov, with values from 0 to 127 representing 0 to -127 dBov. dBov is the level, in decibels, relative to the overload point of the system, i.e. the maximum-amplitude signal that can be handled by the system without clipping. (Note: Representation relative to the overload point of a system is particularly useful for digital implementations, since one does not need to know the relative calibration of the analog circuitry.) For example, in the case of u-law (audio/pcmu) audio [ITU.G.711], the 0 dBov reference would be a square wave with values +/- 8031. (This translates to 6.18 dBm0, relative to u-law's dBm0 definition in Table 6 of G.711.)

6. Signaling Information

The URI for declaring the audio level header extension in an SDP extmap attribute and mapping it to a local extension header identifier is "urn:ietf:params:rtp-hdrext:csrc-audio-level". There is no additional setup information needed for this extension (i.e. no extensionattributes).

An example attribute line in the SDP, for a conference might be:

```
a=extmap:7 urn:ietf:params:rtp-hdrext:csrc-audio-level
```

The above mapping will most often be provided per media stream (in the media-level section(s) of SDP, i.e., after an "m=" line) or globally if there is more than one stream containing audio level indicators in a session.

Presence of the above attribute in the SDP description of a media stream indicates that some or all RTP packets in that stream would contain the audio level information RTP extension header.

Conferencing clients that support audio level indicators and have no mixing capabilities SHOULD always include the direction parameter in the "extmap" attribute setting it to "recvonly". Conference focus entities with mixing capabilities MAY omit the direction or set it to "sendrecv" in SDP offers. Such entities SHOULD set it to "sendonly" in SDP answers to offers with a "recvonly" parameter and to

"sendrecv" when answering other "sendrecv" offers.

The following Figure 4 and Figure 5 show two example offer/answer exchanges between a conferencing client and a focus, and between two conference focus entities.

```
v=0
o=alice 2890844526 2890844526 IN IP6 host.example.com
c=IN IP6 host.example.com
t=0 0
m=audio 49170 RTP/AVP 0 4
a=rtpmap:0 PCMU/8000
a=rtpmap:4 G723/8000
a=extmap:1/recvonly urn:ietf:params:rtp-hdrext:csrc-audio-level

v=0
i=A Seminar on the session description protocol
o=conf-focus 2890844730 2890844730 IN IP6 focus.example.net
c=IN IP6 focus.example.net
t=0 0
m=audio 52543 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=extmap:1/sendonly urn:ietf:params:rtp-hdrext:csrc-audio-level
```

A client-initiated example SDP offer/answer exchange negotiating an audio stream with one-way flow of of audio level information.

Figure 4

```
v=0
i=Un seminaire sur le protocole de description des sessions
o=fr-focus 2890844730 2890844730 IN IP6 focus.fr.example.net
c=IN IP6 focus.fr.example.net
t=0 0
m=audio 49170 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=extmap:1/sendrecv urn:ietf:params:rtp-hdrext:csrc-audio-level

v=0
i=A Seminar on the session description protocol
o=us-focus 2890844526 2890844526 IN IP6 focus.us.example.net
c=IN IP6 focus.us.example.net
t=0 0
m=audio 52543 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=extmap:1/sendrecv urn:ietf:params:rtp-hdrext:csrc-audio-level
```

An example SDP offer/answer exchange between two conference focus entities with mixing capabilities negotiating an audio stream with bidirectional flwo of audio level information.

Figure 5

7. Security Considerations

1. This document defines a means of attributing audio level to a particular participant in a conference. An attacker may try to modify the content of RTP packets in a way that would make audio activity from one participant appear as coming from another.
2. Furthermore, the fact that audio level values would not be protected even in an SRTP session may be of concern in some cases where the activity of a particular participant in a conference is confidential.
3. Both of the above are concerns that stem from the design of the RTP protocol itself and they would probably also apply when using CSRC identifiers the way they were specified in RFC 3550 [RFC3550]. It is therefore important that according to the needs of a particular scenario, implementors and deployers consider use of a lower level security and authentication mechanism.

8. IANA Considerations

This document defines a new extension URI that, if approved, would need to be added to the RTP Compact Header Extensions sub-registry of the Real-Time Transport Protocol (RTP) Parameters registry, according

to the following data:

```
Extension URI: urn:ietf:params:rtp-hdrext:csrc-audio-level
Description:   Mixer-to-client audio level indicators
Contact:       emcho@sip-communicator.org
Reference:     RFC XXXX
```

9. Open Issues

At the time of writing of this document the authors have no clear view on how and if the following list of issues should be address here:

1. Audio levels in video streams. This specification allows use of audio level values in "silent" audio streams that don't otherwise carry any payload thus allowing their delivery within systems where the various focus/mixer components communicate with each other as conference participants. The same train of thought may very well justify audio level transport in video streams.
2. It has been suggested to reference ITU P.56 [ITU.P56.1993] for level measurement. This needs to be investigated.

10. Acknowledgments

Roni Even, Ingemar Johansson, Michael Ramalho and several others provided helpful feedback over the dispatch mailing list.

SIP Communicator's participation in this specification is funded by the NLnet Foundation.

11. Appendix: Design choices

During discussions on the subject of audio levels the decision to transport audio levels in RTP packets, rather than another protocol was questioned several times which is why the authors find it worth explaining here. The following subsections describe alternative mechanisms for delivering audio levels and the reasons why authors decided not to use them.

11.1. SIP event package for conference state

RFC 4575 [RFC4575] defines a conference event package for tightly coupled conferences using the Session Initiation Protocol (SIP) events framework. It allows for the delivery of various conference related details such as conference descriptions, participant count and identity. The document also provides a way of indicating who the

speakers are at any given moment by specifying a mechanism for mapping conference participants to RTP SSRC/CSRC identifiers. All these details are dispatched in an asynchronous manner using the SIP events framework, or, in other words, through NOTIFY SIP requests following an initial SUBSCRIBE from a participant.

Contrary to "plain" active speaker infomation, where significant changes only occur once every several seconds, audio level in human speech is obviously a very time sensitive characteristic which would require frequent updates (i.e. approximately once every 50-100 ms). In order for the update of the user interface to appear "natural" to the user, audio level information would probably have to be delivered for every one or two RTP packets. Using RFC 4575 [RFC4575] or SIP in general for this would generate traffic on the (often low-bandwidth) signalling path comparable to, if not exceeding, the media itself. It may also prove relatively hard for client developers to synchronize the information they receive from SIP messages with the one they obtain from the media flows.

It is probably also worth mentioning that the use of RFC 4575 [RFC4575] for such a feature would make the mechanism incompatible with non-SIP signaling protocols like, for example, XMPP [RFC3920] and its Jingle extensions.

11.2. The RTP Control Protocol (RTCP)

Similar to using SIP, delivering audio levels through RTCP would cause bandidth and synchronization issues. Furthermore the RTP specification [RFC3550] explicitly recommends that the fraction of the session bandwidth added for RTCP be fixed at 5% which could not be sufficient for the transport of audio level indicators.

11.3. Encoding levels in the payload

Given the content specific nature of audio levels, it has been suggested that audio level information be encoded and transmitted as part of the payload. While this is indeed a feasible approach, implementing it would require a substantial effort. In order to implement support for such a feature, client developers would need to explicitly handle it in all individual codec modules of their application. Compared to RTP extensions, the mechanism would therefore represent a substantial additional effort without offering any meaningful advantages.

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.

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

12.2. Informative References

[I-D.lennox-avt-rtp-audio-level-exthdr]
Lennox, J., Ivov, E., and E. Marocco, "A Real-Time Transport Protocol (RTP) Header Extension for Client-to-Mixer Audio Level Indication", draft-lennox-avt-rtp-audio-level-exthdr-02 (work in progress), July 2010.

[ITU.G.711]
International Telecommunications Union, "Pulse Code Modulation (PCM) of Voice Frequencies", ITU-T Recommendation G.711, November 1988.

[ITU.P56.1993]
International Telecommunications Union, "Objective Measurement of Active Speech Level", ITU-T Recommendation P.56, March 1988.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

[RFC3389] Zopf, R., "Real-time Transport Protocol (RTP) Payload for Comfort Noise (CN)", RFC 3389, September 2002.

[RFC3551] Schulzrinne, H. and S. Casner, "RTP Profile for Audio and Video Conferences with Minimal Control", STD 65, RFC 3551, July 2003.

[RFC3920] Saint-Andre, P., Ed., "Extensible Messaging and Presence Protocol (XMPP): Core", RFC 3920, October 2004.

[RFC4353] Rosenberg, J., "A Framework for Conferencing with the Session Initiation Protocol (SIP)", RFC 4353,

February 2006.

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

Authors' Addresses

Emil Ivov (editor)
SIP Communicator
Strasbourg 67000
France

Email: emcho@sip-communicator.org

Enrico Marocco (editor)

Telecom Italia
Via G. Reiss Romoli, 274
Turin 10148
Italy

Email: enrico.marocco@telecomitalia.it

Jonathan Lennox
Vidyo, Inc.
433 Hackensack Avenue
Seventh Floor
Hackensack, NJ 07601
US

Email: jonathan@vidyo.com