An Extension to the Session Description Protocol (SDP) for Media Loopback

draft-hedayat-media-loopback-01.txt

Status of this Memo

By submitting this Internet-Draft, we certify that any applicable patent or other IPR claims of which we are aware have been disclosed and any of which we become aware will be disclosed, in accordance with RFC 3668 (BCP 79).

By submitting this Internet-Draft, we accept the provisions of Section 3 of RFC 3667 (BCP 78).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt
The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

Abstract

Hedayat, et al. Expires April 30, 2005
The wide deployment of VoIP and Video over IP services has introduced new challenges in managing and maintaining voice/video quality, reliability, and overall performance. In particular, media delivery is an area that needs attention. One method of meeting these challenges is monitoring the media delivery performance by looping media back to the transmitter. This is typically referred to as "active monitoring" of services. Media loopback is especially popular in ensuring the quality of transport to the edge of a given VoIP or Video over IP service. Today in networks that deliver real-time media, short of running 'ping' and 'traceroute' to the edge, service providers are left without the necessary tools to actively monitor, manage, and diagnose quality issues with their service. The extension defined herein adds new SDP media attributes which enables establishment of media sessions where the media is looped back to the transmitter. Such media sessions will serve as monitoring and troubleshooting tools by providing the means for measurement of more advanced VoIP and Video Over IP performance metrics.

Table of Contents

1. Introduction..................................................3
2. Terminology...................................................3
3. Offering Entity Behavior.....................................3
4. Answering Entity Behavior....................................4
5. SDP Constructs Syntax........................................4
   5.1 Loopback Type Attribute...................................4
   5.2 Loopback Mode Attribute...................................5
   5.3 Generating the Offer for Loopback Session...............5
   5.4 Generating the Answer for Loopback Session.............6
   5.5 Offerer Processing of the Answer.........................7
   5.6 Modifying the Session....................................7
6. RTP Requirements.............................................7
7. RTCP Requirements...........................................8
8. Examples......................................................8
   8.1 Offer for specific media loopback type...................8
   8.2 Offer for choice of media loopback type...............9
   8.3 Response to INVITE request rejecting loopback media.....10
9. Implementer Guidelines......................................11
10. Security Considerations...................................11
11. IANA Considerations.......................................11
12. Acknowledgements..........................................11
13. References................................................11
    13.1 Normative References................................11
1. Introduction

The overall quality, reliability, and performance of VoIP and Video over IP services relies on the performance and quality of the media path. In order to assure the quality of the delivered media there is a need to monitor the performance of the media transport. One method of monitoring and managing the overall quality of VoIP and Video over IP Services is through monitoring the quality of the media in an active session. This type of "active monitoring" of services is a method of pro-actively managing the performance and quality of VoIP based services.

The goal of active monitoring is to measure the media quality of a VoIP or Video over IP session. A way to achieve this goal is to request an endpoint to loop media back to the other endpoint and to provide media statistics (e.g., RTCP and RTCP XR information). Another method involves deployment of special endpoints that always loop media back for incoming sessions. Although the latter method has been used and is functional, it does not scale to support large networks and introduces new network management challenges. Further, it does not offer the granularity of testing a specific endpoint that may be exhibiting problems.

The extension defined in this memo introduces new SDP media attributes that enable establishment of media sessions where the media is looped back to the transmitter. The offer/answer model per RFC 3264 [RFC3264] is used to establish a loopback connection. Furthermore, this extension provides guidelines on handling RTP (RFC 3550) [RFC3550], as well as usage of RTCP (RFC 3550) [RFC3550] and RTCP XR (RFC 3611) [RFC3611] for reporting media related measurements.

2. Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in RFC 2119 [RFC3264] and indicate requirement levels for compliant implementations.

3. Offering Entity Behavior
An offering entity compliant to this memo and attempting to establish a media session with media loopback MUST include "loopback" media attributes for each individual media description in the offer message. The offering entity MUST look for the "loopback" media attributes in the media description(s) of the response from the answering entity for confirmation that the request is accepted.

4. Answering Entity Behavior

An answering entity compliant to this specification and receiving an offer containing media descriptions with the "loopback" media attributes, MUST acknowledge the request by including the received "loopback" media attributes for each media description in its response. The server MAY reject the "loopback" request for specific media types as defined in section 5.4.1 of this specification.

An answering entity that is not compliant to this specification and which receives an offer with the "loopback" media attributes MAY safely ignore the attribute and treat the incoming offer as a normal request.

5. SDP Constructs Syntax

Two new media attributes are defined: one indicates the type of loopback and one indicates the mode of the loopback.

5.1 Loopback Type Attribute

The loopback type is a property media attribute with the following syntax:

\[ a=loopback:<loopback-type> \]

ABNF for loopback-type (building on RFC 2327):

\[
\text{loopback-type} = \text{loopback-type-choice} \ [ \text{space} \text{loopback-type-choice} ] \\
\text{loopback-type-choice} = "rtp-pkt-loopback" \mid "rtp-media-loopback"
\]

The loopback type is used to indicate the type of loopback. The loopback-type values are rtp-pkt-loopback and rtp-media-loopback.
rtp-pkt-loopback: In this mode, the RTP packets are looped back to the sender at a point before the encoder/decoder function in the receive direction to a point after the encoder/decoder function in the send direction. This effectively re-encapsulates the RTP payload with the RTP/UDP/IP overheads appropriate for sending it in the reverse direction. Any type of encoding related functions, such as packet loss concealment, MUST NOT be part of this type of loopback path.

rtp-media-loopback: This loopback is activated as close as possible to the analog interface and after the decoder so that the RTP packets are subsequently re-encoded prior to transmission back to the sender.

5.2 Loopback Mode Attribute

The loopback mode is a value media attribute that is used to indicate the mode of the loopback. These attributes can be viewed as additional mode attributes similar to sendonly, recvonly, etc. The syntax of the loopback mode media attribute is:

\[
a=\text{loopback-mode}
\]

The loopback-mode values are loopback-source and loopback-mirror.

loopback-source: This attribute specifies that the sender is the media source and expects the receiver to act as a loopback-mirror.

loopback-mirror: This attribute specifies that the receiver will mirror (echo) all received media back to the sender of the RTP stream. No media is generated locally by the reciver for transmission in the mirrored stream.

5.3 Generating the Offer for Loopback Session

If an offerer wishes to make a loopback request, it MUST include both the loopback-type and loopback-mode attribute in a valid SDP offer:

Example:   a=loopback-type:rtp-media-loopback  
a=loopback-source

Note: A loopback offer in a given media description MUST NOT contain the standard mode attributes sendonly, recvonly, sendrecv or inactive.
The offerer may offer more than one loopback-type in the SDP offer. In this case the answer MUST include only one of the loopback types that is accepted by the answerer. The answerer SHOULD give preference to the first loopback-type in the SDP offer.

For loopback-source media (e.g. audio) streams, the port number and the address in the offer indicates where the offerer would like to receive the media stream. The payload type numbers indicate the value of the payload the offerer expects to receive, and would prefer to send. However, the answer might indicate a different payload type number for the same codec. In that case, the offerer MUST send the payload type received in the answer.

5.4 Generating the Answer for Loopback Session

If an answerer wishes to accept the loopback request it MUST include both the loopback mode and loopback type attribute in the answer. If a stream is offered with loopback-source or loopback-mirror attributes, the corresponding stream MUST be loopback-mirror or loopback-source respectively, provided that answerer is capable of supporting the requested loopback-type.

For example, if the offer contains:

```
a=loopback-type:rtp-media-loopback
a=loopback-source
```

The answer that is capable of supporting the offer MUST contain:

```
a=loopback-type:rtp-media-loopback
a=loopback-mirror
```

As previously stated if a stream is offered with multiple loopback type attributes, the corresponding stream MUST contain only one loopback type attribute selected by the answerer.

For example, if the offer contains:

```
a=loopback-type:rtp-media-loopback rtp-pkt-loopback
a=loopback-source
```

The answer that is capable of supporting the offer and chooses to loopback the media using the rtp-media-loopback type MUST contain:

```
a=loopback-type:rtp-media-loopback
```
a=loopback-mirror

5.4.1 Rejecting the Loopback Offer

An offered stream with loopback-source MAY be rejected, if the loopback-type is not specified, the specified loopback-type is not supported, or the endpoint cannot honor the offer for any other reason. The Loopback request may be rejected by setting the media port number to zero, according to RFC 3264 [RFC3264], in the answer.

5.5 Offerer Processing of the Answer

The answer to a loopback-source MUST be loopback-mirror. The answer to a loopback-mirror MUST be loopback-source. In addition, the "m=" line MUST contain at least one codec that the answerer is willing to both send and receive.

If the answer does not contain a=loopback-mirror or a=loopback-source or contains any other standard mode attributes, it is assumed that the loopback extensions are not supported by the target UA.

5.6 Modifying the Session

At any point during the loopback session, either participant may issue a new offer to modify the characteristics of the previous session. In case of SIP this is defined in section 8 of RFC 3264 [RFC3264]. This also includes transitioning from a normal media processing mode to loopback mode, and vice versa.

6. RTP Requirements

An answering entity that is compliant to this specification and accepting a media with rtp-pkt-loopback loopback-type MUST re-generate all of the RTP header fields as it does when transmitting other media. However, the answering entity MUST maintain the timing information of the received RTP packets when generating the RTP timestamp for the transmit packets. Maintaining the timing information of the RTP packets enables the offerer to
re-construct the incoming media and take account for impairments from gaps in the media due to packet loss. Note that RTP Sequence numbers are re-generated by the UAS and will not provide packet loss information to the receiver of the loopback media.

An answering entity that is compliant to this specification and accepting a media with rtp-media-loopback loopback-type MUST transmit all received media back to the sender. The incoming media MUST be treated as if it were to be played (e.g. the media stream MAY receive treatment from PLC algorithms). The answering entity MUST re-generate all the RTP header fields as it would when transmitting media. The UAS MAY choose to encode the loopback media according to any of the media descriptions supported by the UAC. Furthermore, in cases where the same media type is looped back, the UAS MAY choose to preserve number of frames/packet and bitrate of the encoded media according to the received media.

7. RTCP Requirements

The use of the loopback attribute is intended for monitoring of media quality of the session. Consequently the media performance information must be exchanged between the offering and the answering entities. An offering or answering entity that is compliant to this specification MUST support RTCP per [RFC3550] and RTCP-XR per RFC 3611 [RFC3611]. Furthermore, both the client and the server MUST support RTCP-XR Statistics Summary Report Block and VoIP Metric Reports Block per sections 4.6 and 4.7 of RFC 3611 [RFC3611]. The client and the server MAY support other RTCP-XR reporting blocks as defined by RFC 3611 [RFC3611].

8. Examples

This section provides examples for media descriptions using SDP for different scenarios. The examples are given for SIP based transactions and are abbreviated and do not show the complete signaling for convenience.

8.1 Offer for specific media loopback type

A client sends an INVITE request with SDP which looks like:

v=0
o=user1 2890844526 2890842807 IN IP4 126.16.64.4
The client is offering to source the media and expects the server to mirror the RTP stream per rtp-media-loopback loopback type.

A server sends a response with SDP which looks like:

v=0
o=user1 2890844526 2890842807 IN IP4 126.16.64.4
s=Example
i=An example session
e=user@example.com
c=IN IP4 224.2.17.12/127
t=0 0
m=audio 49170 RTP/AVP 0
a=loopback:rtp-media-loopback
a=loopback-mirror

The server is accepting to mirror the media from the client at the media level.

8.2 Offer for choice of media loopback type

A client sends an INVITE request with SDP which looks like:

v=0
o=user1 2890844526 2890842807 IN IP4 126.16.64.4
s=Example
i=An example session
e=user@example.com
c=IN IP4 224.2.17.12/127
t=0 0
m=audio 49170 RTP/AVP 0
a=loopback:rtp-media-loopback rtp-pkt-loopback
a=loopback-source

The client is offering to source the media and expects the server to mirror the RTP stream at either the media or rtp level.

A server sends a response with SDP which looks like:
v=0
c=user1 2890844526 2890842807 IN IP4 126.16.64.4
s=Example
i=An example session
e=user@example.com
c=IN IP4 224.2.17.12/127
t=0 0
m=audio 49170 RTP/AVP 0
a=loopback:rtp-pkt-loopback
a=loopback-mirror

The server is accepting to mirror the media from the client at the packet level.

### 8.3 Response to INVITE request rejecting loopback media

A client sends an INVITE request with SDP which looks like:

v=0
c=user1 2890844526 2890842807 IN IP4 126.16.64.4
s=Example
i=An example session
e=user@example.com
c=IN IP4 224.2.17.12/127
t=0 0
m=audio 49170 RTP/AVP 0
a=loopback:rtp-media-loopback
a=loopback-source

The client is offering to source the media and expects the server to mirror the RTP stream at the media level.

A server sends a response with SDP which looks like:

v=0
c=user1 2890844526 2890842807 IN IP4 126.16.64.4
s=Example
i=An example session
e=user@example.com
c=IN IP4 224.2.17.12/127
t=0 0
m=audio 0 RTP/AVP 0
a=loopback:rtp-media-loopback
a=loopback-mirror
NOTE: Loopback request may be rejected by either not including the loopback mode attribute (for backward compatibility) or setting the media port number to zero, or both, in the response.

9. Implementer Guidelines

This section provides guidelines to the implementers of this extension.

10. Security Considerations

The security considerations of [RFC3261] apply.

11. IANA Considerations

There are no IANA considerations associated with this specification.

12. Acknowledgements

The authors wish to thank Flemming Andreasen, Jeff Bernstein, Paul Kyzivat, and Dave Oran for their comments and suggestions.

13. References

13.1 Normative References


Authors’ Addresses

Kaynam Hedayat
Brix Networks
285 Mill Road
Chelmsford, MA 01824
US

Phone: +1 978 367 5611
EMail: khedayat@brixnet.com
URI: http://www.brixnet.com/

Paul E. Jones
Cisco Systems, Inc.
7025 Kit Creek Rd.
Research Triangle Park, NC 27709
US

Phone: +1 919 392 6948
EMail: paulej@packetizer.com
URI: http://www.cisco.com/

Arjun Roychowdhury
Hughes Software Systems
11717 Exploration Lane
Germantown, MD 20876
US

Phone: +1 301 212 7860
EMail: aroychow@hssworld.com
URI: http://www.hssworld.com/

Chelliah SivaChelvan
Cisco Systems, Inc.
IPR Disclosure Acknowledgement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.
Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Notice

Copyright (C) The Internet Society (2004).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.