

MMUSIC
Internet-Draft
Intended status: Standards Track
Expires: August 28, 2008

T. Ura
K. Oku
NTT
H. Harada
Hitachi
A. Kobayashi
NEC Corp.
M. Stiemerling (Ed.)
NEC Europe Ltd.
February 25, 2008

RTSP 2.0 Asynchronous Notification
draft-stiemerling-rtsp-announce-01

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

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/lid-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on August 28, 2008.

Copyright Notice

Copyright (C) The IETF Trust (2008).

Abstract

Some IPTV deployments that are using the Real Time Streaming Protocol (RTSP) require the ability of the server to notify clients about asynchronous events occurring during an RTSP session. Current deployments typically use the ANNOUNCE method of RTSP 1.0 for sending such asynchronous events from a server to clients by using some proprietary extensions. However, the ANNOUNCE method has been removed from the current RTSP 2.0 draft, leaving the new specification without a mechanism for sending asynchronous messages from the server. This memo describes a use case for such an asynchronous message and proposes a new RTSP 2.0 method.

Table of Contents

- 1. Introduction 3
- 2. Definition of Method 5
 - 2.1. Normative Definitions 5
 - 2.2. Notice Header 6
 - 2.3. Limitations of ANNOUNCE 8
- 3. Feature Tag 9
- 4. Security Considerations 10
- 5. Conclusion 11
- 6. References 12
 - 6.1. Normative References 12
 - 6.2. Informative References 12
- Appendix A. Design Choices for Asynchronous Notifications 13
- Authors' Addresses 14
- Intellectual Property and Copyright Statements 16

1. Introduction

Some IPTV deployments that are using the Real Time Streaming Protocol (RTSP) require the ability of the server to notify clients about asynchronous events occurring during an RTSP session. Such asynchronous events are, for example, end of session, end of stream or change (redirect) of the RTSP server. While redirecting RTSP clients to a different RTSP server is well-described in RTSP 1.0 [RFC2326] and RTSP 2.0 [I-D.ietf-mmusic-rfc2326bis], end of stream or end of session by the server is not defined. This memo aims at changing RTSP 2.0 but not RTSP 1.0.

Some RTSP 1.0 deployments are extending the ANNOUNCE method of [RFC2326], which is defined as:

When sent from client to server, ANNOUNCE posts the description of a presentation or media object identified by the request URL to a server. When sent from server to client, ANNOUNCE updates the session description in real-time.

These implementations use ANNOUNCE to send asynchronous notifications for RTSP sessions from an RTSP server to RTSP clients. The ANNOUNCE method is extended by adding a reason header, to indicate end of stream reason to the clients. However, this extension of the ANNOUNCE method is not standardised.

An example for a deployment using the extended ANNOUNCE method is the Hikari service provided by NTT in Japan [Hikari]. This service is based on the Hikari Service Architecture (HSA), developed by the Hikari Service Architecture Consortium (HSAC, [HSAC]) in Japan. NB: The consortium itself was closed, as the service specification was completed, and the specification is currently used. HSA is actually using the ANNOUNCE method as described above but does not use the client to server mode of it. According to discussions during the IETF#70 meeting, several other deployments (namely cable deployments), are also using ANNOUNCE this way.

The ANNOUNCE method has been removed in RTSP 2.0 [I-D.ietf-mmusic-rfc2326bis], thus also having removed this way of sending asynchronous notifications to clients. The current way in RTSP 2.0 of sending end of stream notifications is using the REDIRECT method (see Section 11.9):

The lack of a Location header in any REDIRECT request is indicative of the server no longer being able to fulfil the current request and having no alternatives for the client to continue with its normal operation. It is akin to a server initiated TEARDOWN that applies both to sessions as well as the

general connection associated with that client.

The REDIRECT request does not include any reason header why the stream or the RTSP session is actually about to end. Furthermore, REDIRECT seems to be logically the wrong place for such a stream or session termination.

Given the lack of a good and extensible semantics in the current RTSP 2.0 draft for sending asynchronous notifications, we propose a new method in the next section. The next section also discusses two different ways of including the new method in the RTSP 2.0 specification.

There was already an attempt to re-introduce the ANNOUNCE method described in [I-D.ietf-mmusic-rtsp-announce]. The goal of this draft was to revive ANNOUNCE for end of stream and change of session description. This memo deliberately reuses some of ideas of [I-D.ietf-mmusic-rtsp-announce] but nothing related to change of session description.

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].

2. Definition of Method

This section defines a new method for RTSP servers to asynchronously ANNOUNCE clients about end of stream or end of RTSP session with a reason code. The new method is called ANNOUNCE, but any other suitable name is also possible (e.g., HALT, STOP, BYE), i.e., it is subject to further discussions.

There are actually two ways of adding this extension to the RTSP protocol. First, the ANNOUNCE method could be added directly to the core RTSP protocol or, second, it could be added as an extension to the core RTSP protocol. This is also subject to WG discussions.

The ANNOUNCE method is a mechanism for RTSP servers to signal RTSP clients about end of stream or end of RTSP session events. ANNOUNCE is an RTSP request that can only be sent from servers to clients, thus requiring a persistent connection between server and client. Otherwise there is no way for the server to send this request method to the client. Therefore, clients are required to keep the RTSP connection to the server open at all times of an RTSP session.

Here is an example RTSP conversation in which an RTSP server announces an end of stream event for a media stream using a non-aggregate URI.

```
S->C: ANNOUNCE rtsp://foo.com/bar.avi/streamid=0 RTSP/2.0
      CSeq: 99
      Session: 12345678
      Notice: 2101 End-Of-Stream
      Range: npt=0-200
      RTP-Info: url=rtsp://foo.com/bar.avi/streamid=0;seq=45102

C->S: RTSP/2.0 200 OK
      CSeq: 99
      Session: 12345678
```

2.1. Normative Definitions

The request-URI of an ANNOUNCE request can be either aggregate or non-aggregate URI.

An ANNOUNCE request must include "CSeq" header and "Notice". It MAY include the following optional headers:

"Range",
"Session",
"RTP-Info".

An ANNOUNCE request MAY include an entity body, in which case it MUST follow the rules for entity body defined in Section 8.2 of [I-D.ietf-mmusic-rfc2326bis]. The entity body can be used to convey further details specific to an event type. If the event type is end-of-stream or session termination announcement, the entity body MAY contain "text/parameter" content type that conveys the reason of the event.

ANNOUNCE does not affect RTSP session state if the event type is "2101 End of Stream" but does affect the RTSP session state if the event type is "3000 End of Session". If a receiver does not understand any of the headers in an ANNOUNCE request, it simply ignores those headers.

The next section defines a new RTSP headers for ANNOUNCE method: "Notice".

2.2. Notice Header

This section defines a new mandatory header. The Notice header is identifying the type of event pertaining to the ANNOUNCE request.

The Notice header is defined in ABNF as:

```
Notice           = "Notice" ":" Notice
Notice           = Notice-code SP Notice-string
Notice-code      = 4DIGITS
Notice-string    = token
```

where:

-- token is defined in section 17 of [RTSP_NEW].

The "Notice" header applies only the ANNOUNCE method, which is sent from server to client.

The following pairs for Notice-code and Notice-string are defined in this memo.

Notice-code	Notice-string	Description
1103	Playout Stalled	-/-
1104	Playout Resumed	Temporarily stopped
2101	End-of-Stream Reached	Content terminated
2103	Transition	In transition
2104	Start-of-Stream Reached	Returned to the initial content
2306	Continuous Feed Terminated	Live finished
2401	Ticket Expired	Viewing right expired
4400	Error Reading Content Data	Data read error
5200	Server Resource Unavailable	Resource cannot be obtained
5401	Downstream Failure	Stream could not be obtained
5402	Client Session Terminated	-/-
5403	Server Shutting Down	-/-
5404	Internal Server Error	-/-
5501	End-of-Window_term	-/-
5502	End-of-Contract_term	-/-

Table 1: The Notice values

If "Notice" is "2101 End-Of-Stream", the optional RTP-Info header SHOULD contain the "seq" attribute that indicates the sequence number of the next RTP packet.

2.3. Limitations of ANNOUNCE

The server to client ANNOUNCE method is issued only if the server has the means to contact the client when it has information to push. This may not be possible if the RTSP connection between server and client is not persistent. In such cases, the server will simply skip the sending of ANNOUNCE requests. The server MUST NOT queue up the ANNOUNCE requests to be sent when client eventually connects. Such a queue would unnecessarily complicate server implementations.

3. Feature Tag

The support of the ANNOUNCE method is represented by this feature tag:

method.announce

This feature tag applies to both servers and proxies.

Implementations claiming "method.announce" feature tag MUST support the new "Notice" header defined in previous section.

4. Security Considerations

This initial version of this memo does not have yet any security considerations, but they will be added with the next revision.

5. Conclusion

This memo is work in progress and is requesting feedback from the MMUSIC working group. The first version of this draft has been presented and discussed during the IETF370 meeting. There has been no final conclusion during and after the meeting on how to proceed w.r.t. to end of stream and end of session notifications.

During the IETF#70 meeting three different "semantics" for ANNOUNCE have been discussed:

- o notification of end of stream; This should be incorporated, may be even in as an explicit method due to implementers. There was additional support for having this kind of notifications.
- o "session-type" notifications: "sorry, cant serve you anymore": It is believed that this is covered well in ther current specification.
- o move an RTSP session from one server host to another: well-done by the current specification.

This memo needs to be further discussed to clarify these open issues:

- o Should the proposed mechanism can be integrated in the current RTSP 2.0 specification or if it should be defined as an extension to the current RTSP 2.0 specification?
- o The text mentions just RTSP server, but never clarifies the role of ANNOUNCE w.r.t. RTSP proxies;
- o Is the name ANNOUNCE appropriate for this method?
- o The entity body should be better specified
- o The only real reason codes are not very helpful yet.
- o Exemplifying use cases are missing
- o Extensibility section is missing
- o IANA section is missing.

6. References

6.1. Normative References

- [I-D.ietf-mmusic-rfc2326bis]
Schulzrinne, H., Rao, A., Lanphier, R., Westerlund, M.,
and M. Stiemerling, "Real Time Streaming Protocol 2.0
(RTSP)", draft-ietf-mmusic-rfc2326bis-16 (work in
progress), November 2007.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
Requirement Levels", BCP 14, RFC 2119, March 1997.

6.2. Informative References

- [HSAC] "Hikari Service Architecture Consortium", Web Site [http://
web.archive.org/web/20040518075704/http://
www.hikari-sac.org/](http://web.archive.org/web/20040518075704/http://www.hikari-sac.org/), October 2007.
- [Hikari] "Hikari Service Architecture", Web Site [http://
www.itu.int/itudoc/itu-t/com13/ipexpert/ipmedia/
71304_pp7.ppt](http://www.itu.int/itudoc/itu-t/com13/ipexpert/ipmedia/71304_pp7.ppt), October 2007.
- [I-D.ietf-mmusic-rtsp-announce]
Zeng, T., "RTSP Announce Method",
draft-ietf-mmusic-rtsp-announce-01 (work in progress),
February 2005.
- [RFC2326] Schulzrinne, H., Rao, A., and R. Lanphier, "Real Time
Streaming Protocol (RTSP)", RFC 2326, April 1998.

Appendix A. Design Choices for Asynchronous Notifications

Appendix A of [I-D.ietf-mmusic-rtsp-announce] discusses several design choices for implementing the ANNOUNCE method semantics with other methods of RTSP and also RTP. We deem the arguments described as still valid and also applicable for the in this memo discussing the ANNOUNCE method.

Authors' Addresses

Tetsuya Ura
Nippon Telegraph and Telephone Corporation
1-1 Hikarinooka
Yokosuka-Shi Kanagawa 239-0847
Japan

Phone: +81 46 859 3780
Email: ura.tetsuya@lab.ntt.co.jp

Kenshin Oku
Nippon Telegraph and Telephone Corporation
1-1 Hikarinooka
Yokosuka-Shi Kanagawa 239-0847
Japan

Phone: +81 46 859 2528
Email: oku.kenshin@lab.ntt.co.jp

Hiromi Harada
Hitachi, Ltd.
890 Kashimada
Saiwai-ku Kawasaki-Shi Kanagawa 212-8567
Japan

Phone: +81 44 549 1578
Email: hiromi.harada.jv@hitachi.com

Akira Kobayashi
NEC Corporation
11-5 Shibaura 2-chone
Minato-ku, Tokyo 108-8557
Japan

Phone: +81 3 5476 1084
Email: a-kobayasi@ce.jp.nec.com

Martin Stiemerling
NEC Laboratories Europe
Kurfuerstenanlage 36
Heidelberg 69115
Germany

Phone: +49 6221 4342 113
Fax: +49 6221 4342 155
Email: stiemerling@nw.neclab.eu
URI: <http://www.netlab.nec.de/>

Full Copyright Statement

Copyright (C) The IETF Trust (2008).

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.

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, THE IETF TRUST 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.

Intellectual Property

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.

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).

