SIPREC
Internet-Draft
Intended status: Informational
Expires: August 31, 2014

A. Hutton, Ed.
Unify
L. Portman, Ed.
NICE Systems
R. Jain
IPC Systems
K. Rehor
Cisco Systems, Inc.
February 27, 2014

An Architecture for Media Recording using the Session Initiation Protocol
draft-ietf-siprec-architecture-12

Abstract

Session recording is a critical requirement in many communications environments such as call centers and financial trading. In some of these environments, all calls must be recorded for regulatory, compliance, and consumer protection reasons. Recording of a session is typically performed by sending a copy of a media stream to a recording device. This document describes architectures for deploying session recording solutions in an environment which is based on the Session Initiation Protocol (SIP).

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Session recording is a critical requirement in many communications environments such as call centers and financial trading. In some of these environments, all calls must be recorded for regulatory, compliance, and consumer protection reasons. Recording of a session is typically performed by sending a copy of a media stream to a recording device. This document describes architectures for deploying session recording solutions as defined in "Use Cases and Requirements for SIP-Based Media Recording (SIPREC)" [RFC6341].

This document focuses on how sessions are established between a Session Recording Client (SRC) and the Session Recording Server (SRS) for the purpose of conveying the Replicated Media and Recording Metadata (e.g. Identity of parties involved) relating to the Communication Session.

Once the Replicated Media and Recording Metadata have been received by the SRS they will typically be archived for retrieval at a later time. The procedures relating to the archiving and retrieval of this information is outside the scope of this document.

This document only considers active recording, where the SRC purposefully streams media to a SRS. Passive recording, where a recording device detects media directly from the network (E.g. using port mirroring techniques), is outside the scope of this document. In addition, lawful intercept is outside the scope of this document which takes account of the IETF policy on wiretapping [RFC2804].

The Recording Session that is established between the SRC and the SRS uses the normal procedures for establishing INVITE initiated dialogs as specified in [RFC3261] and uses SDP for describing the media to be used during the session as specified in [RFC4566]. However it is intended that some extensions to SIP (E.g. Headers, Option Tags, Etc.) will be defined to support the requirements for media recording. The Replicated Media is required to be sent in real-time to the SRS and is not buffered by the SRC to allow for real-time analysis of the media by the SRS.

2. Definitions

Session Recording Server (SRS): A Session Recording Server (SRS) is a SIP User Agent (UA) that is a specialized media server or collector that acts as the sink of the recorded media. An SRS is typically implemented as a multi-port device that is capable of receiving media from multiple sources simultaneously. An SRS is the sink of the communication session metadata.

Session Recording Client (SRC): A Session Recording Client (SRC) is a SIP User Agent (UA) that acts as the source of the recorded media, sending it to the SRS. An SRC is a logical function. Its capabilities may be implemented across one or more physical devices. In practice, an SRC could be a personal device (such as a SIP phone), a SIP Media Gateway (MG), a Session Border Controller (SBC) or a SIP Media Server (MS) integrated with an Application Server (AS). This specification defines the term SRC such that all such SIP entities can be generically addressed under one definition. The SRC provides comunication session metadata to the SRS.

Communication Session (CS): A session created between two or more SIP User Agents (UAs) that is the subject of recording.

Recording Session (RS): The SIP session created between an SRC and SRS for the purpose of recording a CS.

Recording aware User Agent (UA): A SIP User Agent that is aware of SIP extensions associated with the CS. Such extensions may be used to notify the Recording aware UA that a session is being recorded, or by a Recording aware UA to express preferences as to whether a recording should be started, paused, resumed or stopped.

Recording unaware User Agent (UA): A SIP User Agent that is unaware of SIP extensions associated with the CS. Such Recording unaware UA will be notified that a session is being recorded or express preferences as to whether a recording should be started, paused, resumed or stopped via some other means that is out of scope for the SIP media recording architecture.

Recording Metadata: The metadata describing the CS that is required by the SRS. This will include for example the identity of users that participate in the CS and dialog state. Typically this metadata is archived with the Replicated Media at the SRS. The recording metadata is delivered in real-time to the SRS.

Replicated Media: A copy of the media associated with the CS created by the SRC and sent to the SRS. It may contain all the media associated with the CS (e.g. Audio and Video) or just a subset (e.g. Audio). Replicated Media is part of Recording Session.

3. Session Recording Architecture

3.1. Location of the SRC

This section contains some example session recording architectures showing how the SRC is a logical function that can be located in or split between various physical components.

3.1.1. B2BUA acts as a SRC

A SIP Back to Back User Agent (B2BUA) which has access to the media to be recorded may act as an SRC. The B2BUA may already be aware that a session needs to be recorded before the initial establishment of the CS or the decision to record the session may occur after the session has been established.

If the SRC makes the decision to initiate the RS, then it will initiate the establishment of a SIP RS by sending an INVITE to the SRS.

If the SRS makes the decision to initiate the recording session, then it will initiate the establishment of a SIP RS by sending an INVITE to the SRC.

The RS INVITE contains information which identifies the session as being established for the purposes of recording and prevents the session from being accidentally rerouted to a UA which is not an SRS if the RS was initiated by SRC or vice-versa.

The B2BUA/SRC is responsible for notifying the UAs involved in the CS that the session is being recorded.

The B2BUA/SRC is responsible for complying with requests from recording aware UAs or through some configured policies indicating that the CS should not be recorded.

```
                                              +-----------+
                          (Recording Session) |  Session  |
                             +------SIP------>| Recording |
                             |                |  Server   |
                             |  +--RTP/RTCP-->|  (SRS)    |
                             |  |             +-----------+
                             V  V                  ^
                        +-------------+            |
                        |             |            |
                        |             |-- Metadata -+
                        |             |
                        |    B2BUA    |
                        |             |
                        |   Session   |
     +--------+         |  Recording  |         +---------+
     |        |<- SIP ->|   Client    |<- SIP ->|         |
     |  UA-A  |         |   (SRC)     |         |  UA-B   |
     |        |<- RTP/->|             |<- RTP/->|         |
     +--------+   RTCP  |             |   RTCP  +---------+
                        +-------------+
     |____________________________________________________|
                    (Communication Session)
```

Figure 1: B2BUA Acts as the Session Recording Client.

3.1.2. Endpoint acts as SRC

A SIP Endpoint / UA may act as a SRC. in which case the endpoint sends the Replicated Media to the SRS.

If the endpoint makes the decision to initiate the Recording Session then it will initiate the establishment of a SIP Session by sending an INVITE to the SRS.

If the SRS makes the decision to initiate the Recording Session then it will initiate the establishment of a SIP Session by sending an INVITE to the endpoint. The actual decision mechanism is out of scope for the SIP media recording architecture.

```
          (Recording Session) +-----------+
         +----------SIP------>|           |
         |  +----RTP/RTCP---->|  Session  |
         |  |                 | Recording |
         |  |                 |  Server   |
         |  | +-- Metadata -->|   (SRS)   |
         |  | |               |           |
         |  | |               +-----------+
         |  | |
         |  | |
         |  | |
         |  | |
         V  V |  (Communication Session)
      +--+------+                     +---------+
      |         |<-------SIP--------->|         |
      |  UA-A   |                     |  UA-B   |
      |  (SRC)  |<-----RTP/RTCP------>|         |
      +---------+                     +---------+
```

Figure 2: SIP Endpoint acts as the Session Recording Client

3.1.3. A SIP Proxy cannot be a SRC

A SIP Proxy is unable to act as an SRC because it does not have access to the media and therefore has no way of enabling the delivery of the replicated media to the SRS.

3.1.4. Interaction with MEDIACTRL

The MEDIACTRL architecture [RFC5567] describes an architecture in which an Application Server (AS) controls a Media Server (MS) which may be used for purposes such as conferencing and recording media streams. In the [RFC5567] architecure the AS typically uses SIP Third Party Call Control (3PCC) to instruct the SIP UAs to direct their media to the Media Server.

The SRC or the SRS described in this document may be architected according to [RFC5567]; and therefore, when further decomposed, they may be made up of an application server (AS) which uses a mediactrl interface to control a media server (MS).

As shown in figure 3, when the SRS is architected according to [RFC5567] the MS acts as a sink of the recording media and the AS acts as a sink of the metadata and the termination point for RS SIP signaling. As shown in figure 4, when the SRC is architected according to [RFC5567] the MS acts as a source of recording media and the AS acts as a source of the metadata and the termination point for RS SIP signaling.

```
                                     Session Recording Server (SRS)
                              +----------------------------------------+
                              |                                        |
          (Recording Session) |  +-----------+          +------------+ |
          +------------SIP----|->|           |          |            | |
          |                   |  | MEDIACTRL |MEDIACTRL |   Media    | |
          |                   |  |Application|<-------->|   Server   | |
          |    +-----Metadata--->|  Server   |          |  (Recorder)| |
          |    |              |  |           |          |            | |
          |    |              |  +-----------+          +------------+ |
          |    |              |                              ^         |
          |    |              +------------------------------|---------+
          |    |  +--------------- RTP/RTCP -----------------+
          |    |  |
          V    |  V
        +---+------+                          +---------+
        |          |<-------SIP-------------->|         |
        |   UA-A   | (Communication Session)  |  UA-B   |
        |   (SRC)  |<-------RTP/RTCP--------->|         |
        +----------+                          +---------+
```

Figure 3: Example of Session Recording Server using MEDIACTRL

```
                                                    +----------+
                 (Recording Session)                | Session  |
           +-----------SIP------------------------->|Recording |
           | +----------Metadata------------------->|  Server  |
           | |                                      |   (SRS)  |
           V | UA-A Session Recording Client (SRC)  +----------+
    +----------------------------------------+         ^
    |                                        |         |
    |  +-----------+          +------------+ |         |
    |  |           | Control  |            |<-RTP/RTCP-+    +---------+
    |  |    UA     | Protocol |   Media    | |              |         |
    |  |Application|<-------->|  Server    | |<----SIP----->|  UA-B   |
    |  |  Server   |          |            |<-----RTP------>|         |
    |  |           |          |            | |              +---------+
    |  +-----------+          +------------+ |
    |                                        |
    +----------------------------------------+
```

Figure 4: Example of Session Recording Client decomposition

3.1.5. Interaction with Conference Focus

In the case of a centralised conference a combination of the conference focus and mixer [RFC4353] may act as a SRC and therefore provide the SRS with the replicated media and associated recording metadata. In this arrangement the SRC is able to provide media and metadata relating to each of the participants, including, for example, any side conversations where the media passes through the mixer.

Conference Focus can either provide mixed replicated media or separate streams per conference participant (as depicted in the Figure 5).

The conference focus may also act as a Recording Aware UA in the case when one of the participants acts as a SRC.

In an alternative arrangement a SIP endpoint which is a conference participant can act as an SRC. The SRC will in this case have access to the media and metadata relating to that particular participant and may be able to obtain additional metadata from the conference focus. The SRC may for example use the conference event package as described in [RFC4575] to obtain information about other participants which it provides to the SRS within the recording metadata.

The SRC may be involved in the conference from the very beginning or may join at some later point of time.

```
                                User 1
                            +-----------+
                            |           |
                            |           |
                            |Participant|
                            |     1     |
                            |           |
                            +-----------+
                                ^ ^SIP
                            RTP | |Dialog
                                | |1
       User 2                   V V       Recording
    +-----------+           +-----------+  Session     *************
    |           |           |           |<------------->*           *
    |           |<-- RTP -->|           |<-RTP/RTCP 1->*           *
    |Participant|<--------->| Focus/SRC |<-RTP/RTCP 2->*    SRS    *
    |     2     |  SIP      |           |<-RTP/RTCP 3->*           *
    |           |  Dialog   |           |              *           *
    +-----------+  2        +-----------+              *************
                                 ^ ^
                                 | |SIP
                             RTP | |Dialog
                                 | |3
                                 V V
                            +-----------+
                            |           |
                            |           |
                            |Participant|
                            |    3      |
                            |           |
                            +-----------+
                               User 3
```

Figure 5: Conference Focus acting as an SRC.

3.2. Establishing the Recording Session

The SRC or the SRS may initiate the Recording Session.

It should be noted that the Recording Session is independent from the CS that is being recorded at both the SIP dialog level and at the session level.

Concerning media negotiation, regular SIP/SDP capabilities should be used, and existing transcoding capabilities and media encryption should not be precluded.

3.2.1. SRC Initiated Recording

When the SRC initiates the Recording Session for the purpose of conveying media to the SRS it performs the following actions:

- The SRC is provisioned with a Unified Resource Identifier (URI) for the SRS, which is resolved through normal [RFC3263] procedures.

- Initiates the dialog by sending an INVITE request to the SRS. The dialog is established according to the normal procedures for establishing an INVITE initiated dialog as specified in [RFC3261].

- Include in the INVITE an indication that the session is established for the purpose of recording the associated media.

- If the Replicated Media is to be started immediately then the SRC will include an SDP attribute of "a=sendonly" for each media line or "a=inactive" if it is not ready to transmit the media.

- The Recording Session may replicate all media associated with the CS or only a subset.

- Replicates the media streams that are to be recorded and transmits the media to the SRS.

3.2.2. SRS Initiated Recording

When the SRS initiates the media recording session with the SRC it performs the following actions:

- The SRS is provisioned with a Unified Resource Identifier (URI) for the SRC, which is resolved through normal [RFC3263] procedures.

- Sends an INVITE request to the SRC.

- Includes in the INVITE an indication that the session is established for the purpose of recording the associated media.

- Identifies the sessions that are to be recorded. The actual mechanism of the identification depends on SRC policy.

- If the Recording Session is to be started immediately then the SRS will include an SDP attribute of "a=recvonly" for each media line or "a=inactive" if it is not ready to receive the media.

If the SRS does not have prior knowledge of what media streams are available to be recorded it can make use of an offerless INVITE which allows the SRC to make the initial Session Description Protocol (SDP) offer.

3.2.3. Pause/Resume Recording Session

The SRS or the SRC may pause the recording by changing the SDP direction attribute to "inactive" and resume the recording by changing the direction back to "recvonly" or "sendonly".

3.2.4. Media Stream Mixing

In a basic session involving only audio there are typically two audio /RTP streams between the two UAs involved transporting media in each direction. When recording this media, the two streams may be mixed or not mixed at the SRC before being transmitted to the SRS. In the case when they are not mixed, two separate streams are sent to the SRS. In the mixed case, a single mixed media stream is sent to the SRS. However, in the case when the media streams are not mixed, the SDP offer sent to the SRS must describe two separate media streams.

3.2.5. Media Transcoding

The CS and the RS are negotiated separately using the standard SDP offer/answer exchange which may result in the SRC having to perform media transcoding between the two sessions. If the SRC is not capable of performing media transcoding it may limit the media formats in the offer to the SRS depending on what media is negotiated on the CS or may limit what it includes in the offer on the CS if it has prior knowledge of the media formats supported by the SRS. However typically the SRS will be a more capable device which can provide a wide range of media format options to the SRC and may also be able to make use of a media transcoder as detailed in [RFC5369].

3.2.6. Lossless Recording

Session recording may be a regulatory requirement in certain communication environments. Such environments may impose a requirement generally known as Lossless Recording. An overall lossless recordingsolution may involve multiple layers of solutions. Individual aspects of the solutions may range from administering networks for appropriate QoS, reliable transmission of recorded media and perhaps certain SIPREC protocol level capabilities in SRC and SRS.

3.3. Recording Metadata

3.3.1. Contents of recording metadata

The metadata model is defined in [I-D.ietf-siprec-metadata].

3.3.2. Mechanisms for delivery of metadata to SRS

The SRS obtains session recording metadata from the SRC. The metadata is transported via SIP based mechanisms as specified in [I-D.ietf-siprec-protocol]

It is also possible that metadata is transported via non SIP based mechanisms but these are considered out of scope.

It is also possible to have RS session without the metadata, in such case SRS will be receiving it by some other means or not at all.

3.4. Notifications to the Recorded User Agents

Typically a user that is involved in a session that is to be recorded is notified by an announcement at the beginning of the session or may receive some warning tones within the media. However the standardization of media recording protocols when using SIP enable an indication that the call is being recorded to be included in the SIP requests and responses associated with that CS.

It is the SRC that provides the notification to all SIP UAs for which it is replicating received media for the purpose of recording including the local user if the SRC is a SIP endpoint.

3.5. Preventing the recording of a SIP session

A Recording Aware UA may during the initial session establishment or during an established session provide an indication of their preference with regard to recording the media in the CS. The mechanism for this are specified in [I-D.ietf-siprec-protocol]

4. IANA considerations

This document has no actions for IANA. This draft mentions SIP/SDP extensions. The associated IANA considerations are addressed in [I-D.ietf-siprec-protocol] that defines them.

5. Security considerations

The Recording Session is fundamentally a standard SIP dialog and media session and therefore makes use of existing SIP security mechanisms for securing the Recording Session and Recording Metadata.

The intended use of this architecture is only for the case where the users are aware that they are being recorded, and the architecture provides the means for the SRC to notify users that they are being recorded.

This architectural solution is not intended to support lawful intercept which in contrast requires that users are not informed.

It is the responsibility of the SRS to protect the Replicated Media and Recording Metadata once it has been received and archived. The stored content must be protected using a cipher at least as strong (or stronger) than the original content however the mechanism for protecting the storage and retrieval from the SRS is out of scope of this work. The keys used to store the data must also be securely maintained by the SRS and should only be released, securely, to authorized parties. How to secure these keys, properly authorize a receiving party, or securely distribute the keying material is also out of scope of this work.

Protection of the RS should not be weaker than protection of the CS, and may need to be stronger because the media is retransmitted (allowing more possibility for interception). This applies to both the signaling and media paths.

It is essential that the SRC will authenticate the SRS because the client must be certain that it is recording on the right recording system. It is less important that the SRS authenticate the SRC, but implementations must have the ability to perform mutual authentication.

In some environments, it is desirable to not decrypt and re-encrypt the media. This means the same media encryption key is negotiated and used within the CS and RS. If for any reason the media are decrypted on the CS, and are re-encrypted on the RS, a new key must be used.

The retrieval mechanism for media recorded by this protocol is out of scope. Implementations of retrieval mechanisms should consider the security implications carefully as the retriever is not usually a party to the call that was recorded. Retrievers should be authenticated carefully. The crypto suites on the retrieval should be no less strong than used on the RS, and may need to be stronger.

6. Acknowledgements

Thanks to John Elwell, Brian Rosen, Alan Johnson, Cullen Jennings, Hadriel Kaplan, Henry Lum, Paul Kyzivat, Parthasarathi R, Ram Mohan R, Charles Eckel, Friso Feenstra and Dave Higton for their significant contributions and assistance with this document and Working Group, and to all the members of SIPREC WG mailing list for providing valuable input to this work.

7. Informative References

[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.

[RFC3263] Rosenberg, J. and H. Schulzrinne, "Session Initiation Protocol (SIP): Locating SIP Servers", RFC 3263, June 2002.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, July 2006.

[RFC6341] Rehor, K., Portman, L., Hutton, A., and R. Jain, "Use Cases and Requirements for SIP-Based Media Recording (SIPREC)", RFC 6341, August 2011.

[I-D.ietf-siprec-metadata] R, R., Ravindran, P., and P. Kyzivat, "Session Initiation Protocol (SIP) Recording Metadata", draft-ietf-siprec-metadata-15 (work in progress), February 2014.

[I-D.ietf-siprec-protocol] Portman, L., Lum, H., Eckel, C., Johnston, A., and A. Hutton, "Session Recording Protocol", draft-ietf-siprec-protocol-12 (work in progress), February 2014.

[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.

[RFC5567] Melanchuk, T., "An Architectural Framework for Media Server Control", RFC 5567, June 2009.

[RFC5369] Camarillo, G., "Framework for Transcoding with the Session Initiation Protocol (SIP)", RFC 5369, October 2008.

[RFC2804] IAB and IESG, "IETF Policy on Wiretapping", RFC 2804, May 2000.

Authors’ Addresses

Andrew Hutton (editor)
Unify
Hofmannstrasse 51
81359 Munich
Germany

Email: andrew.hutton@unify.com

Leon Portman (editor)
NICE Systems
8 Hapnina
Ra’anana 43017
Israel

Email: leon.portman@gmail.com

Rajnish Jain
IPC Systems
777 Commerce Drive
Fairfield, CT 06825
USA

Email: rajnish.jain@outlook.com

Ken Rehor
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA

Email: krehor@cisco.com