SIPREC A. Hutton, Ed.
Internet-Draft Unify
Intended status: Informational L. Portman, Ed.
Expires: August 31, 2014 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:

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

- o 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].

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

- o 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.

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

- o 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:

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

- o Sends an INVITE request to the SRC.

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

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

- o 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

SIPREC K. Rehor, Ed.
Internet-Draft Cisco Systems
Intended status: Informational L. Portman, Ed.
Expires: December 11, 2011 NICE Systems
A. Hutton
Siemens Enterprise
Communications
R. Jain
IPC Systems
June 09, 2011

Use Cases and Requirements for SIP-based Media Recording (SIPREC)
draft-ietf-siprec-req-12

Abstract

Session recording is a critical requirement in many business communications environments such as call centers and financial trading floors. In some of these environments, all calls must be recorded for regulatory and compliance reasons. In others, calls may be recorded for quality control or business analytics.

Recording is typically performed by sending a copy of the session media to the recording devices. This document specifies requirements for extensions to SIP that will manage delivery of RTP media to a recording device. This is being referred to as SIP-based Media Recording.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

Session recording is a critical operational requirement in many businesses, especially where voice is used as a medium for commerce and customer support. A prime example where voice is used for trade is the financial industry. The call recording requirements in this industry are quite stringent. The recorded calls are used for dispute resolution and compliance. Other businesses such as customer support call centers typically employ call recording for quality control or business analytics, with different requirements.

Depending on the country and its regulatory requirements, financial trading floors typically must record all calls. In contrast, call centers typically only record a subset of the calls, and calls must not fail regardless of the availability of the recording device.

Respecting the privacy rights and wishes of users engaged in a call is of paramount importance. In many jurisdictions participants have a right to know that the session is being recorded or might be recorded, and have a right to opt out, either by terminating the call or by demanding that the call not be recorded. Therefore this document contains requirements for being able to notify users that a call is being recorded and for users to be able to request that a call not be recorded. Use cases where users participating in a call are not informed that the call is or might be recorded are outside the scope of this document. In particular, lawful intercept is outside the scope of this document.

Furthermore, one-size-fits-all model will not fit all markets where the scale and cost burdens vary widely having different needs for solution capabilities such as media injection, transcoding, and security. If a standardized solution supports all of the requirements from every recording market, but doing so would be expensive for markets with lesser needs, then proprietary solutions for those markets will continue to propagate. Care must be taken, therefore, to make a standards-based solution support optionality and flexibility.

This document specifies requirements for using SIP [RFC3261] between a Session Recording Client and a Session Recording Server to control the recording of media that has been transmitted in the context of a Communication Session. A Communication Session is the "call" between participants. The Session Recording Client is the source of the recorded media. The Session Recording Server is the sink of recorded media. It should be noted that the requirements for the protocol between a Session Recording Server and Session Recording Client have very similar requirements (such as codec and transport negotiation, encryption key interchange, firewall traversal) as compared to regular SIP media sessions. The choice of SIP for session recording provides reuse of an existing protocol.

The recorded sessions can be any RTP media sessions including voice, DTMF (as defined by [RFC4733]), video, and text (as defined by [RFC4103]).

An archived session recording is typically comprised of the Communication Session media content and the Communication Session Metadata. The Communication Session Metadata allows recording archives to be searched and filtered at a later time and allows a session to be played back in a meaningful way, e.g., with correct synchronization between the media. The Communication Session Metadata needs to be conveyed from the Session Recording Client to the Session Recording Server.

This document only considers active recording, where the Session Recording Client purposefully streams media to a Session Recording Server. Passive recording, where a recording device detects media directly from the network, is outside the scope of this document.

2. Requirements notation

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] and indicate requirement levels for compliant mechanisms.

3. 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 recorded 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 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 Communication Session.

Figure 1 pictorially represents the relationship between a Recording Session and Communication Session.

```
  +-------------+                                     +-----------+
  |             |        Communication Session        |           |
  |      A      |<----------------------------------->|     B     |
  |             |                                     |           |
  +-------------+                                     +-----------+
  ...................................................................
  .                            Session                              .
  .                           Recording                             .
  .                            Client                               .
  ...................................................................
                                |
                                | Recording
                                | Session
                                |
                                v
                          +------------+
                          |   Session  |
                          |  Recording |
                          |   Server   |
                          +------------+
```

Figure 1

Metadata: Information that describes recorded media and the CS to which they relate.

Pause and Resume during a Communication Session: Pause: The action of temporarily discontinuing the transmission and collection of RS media Resume: The action of recommencing the transmission and collection of RS media

Most security-related terms in this document are to be understood in the sense defined in [RFC4949]; such terms include, but are not limited to, "authentication", "confidentiality", "encryption", "identity", and "integrity".

4. Use Cases

Use Case 1: Full-time Recording: One Recording Session for each Communication Session.

For example, the diagram below shows the lifecycle of Communication Sessions (CS) and the relationship to the Recording Sessions (RS)

```
CS  |--- CS 1 ---|      |--- CS 2 ---|     |--- CS 3 ---|

RS  |--- RS 1 ---|      |--- RS 2 ---|     |--- RS 3 ---|
t--->
```

Figure 2

Record every CS for specific extension/person.

The need to record all calls is typically due to business process purposes (such as transaction confirmation or dispute resolution) or to ensure compliance with governmental regulations. Applications include enterprise, contact center, and financial trading floors.

Also commonly known as Total Recording.

Use Case 2: Selective Recording: Start a Recording Session when a Communication Session to be recorded is established.

In this example, Communication Sessions 1 and 3 are recorded but CS 2 is not.

```
CS  |--- CS 1 ---|      |--- CS 2 ---|     |--- CS 3 ---|

RS  |--- RS 1----|                         |--- RS 2 ---|
t--->
```

Figure 3

Use Case 3: Start/Stop a Recording Session during a Communication Session.

The Recording Session starts during a Communication Session, either manually via a user-controlled mechanism (e.g. button on user's phone) or automatically via an application (e.g. a Contact Center customer service application) or business event. A Recording Session either ends during the Communication Session, or when the

Communication Session ends. One or more Recording Sessions may record each Communication Session.

```
CS  |------------- Communication Session -----------|

RS            |---- RS 1 ----|  |---- RS 2 -----|
t--->
```

Figure 4

Use Case 4: Persistent Recording: A single Recording Session captures one or more Communication Sessions.

```
          |--- CS 1 ---|      |--- CS 2 ---|     |--- CS 3 ---|

RS  |---------------------- Recording Session ---------------------|
t--->
```

Figure 5

A Recording Session records continuously without interruption. Periods when there is no CS in progress must be reproduced upon playback (e.g. by recording silence during such periods or by not recording such periods but marking them by means of metadata for utilization on playback, etc.). Applications include financial trading desks and emergency (first-responder) service bureaus. The length of a Persistent Recording Session is independent from the length of the actual Communication Sessions. Persistent Recording Sessions avoid issues such as media clipping that can occur due to delays in Recording Session establishment.

The connection and attributes of media in the Recording Session are not dynamically signaled for each Communication Session before it can be recorded; however, codec re-negotiation is possible.

In some cases, more than one concurrent Communication Session (on a single end-user apparatus, e.g. trading floor turret) is mixed into one Recording Session:

```
              |-------- CS 1 -------|
                 |-------- CS 2 -------|
            |-------- CS 3 -------|

RS  |----------- Recording Session --------------|
t--->
```

Figure 6

Use Case 5: Real-time Recording Controls.

For an active Recording Session, privacy or security reasons may demand not capturing a specific portion of a conversation. An example is for PCI (payment card industry) compliance where credit card info must be protected. One solution is to not record a caller speaking their credit card information.

An example of a real-time controls is Pause/Resume.

Use Case 6: IVR / Voice Portal Recording.

Self-service Interactive Voice Response applications may need to be recorded for application performance tuning or to meet compliance requirements.

Metadata about an IVR session recording must include session information and may include application context information (e.g. VoiceXML session variables, dialog names, etc.)

Use Case 7: Enterprise Mobility Recording.

Many agents and enterprise workers whose calls are to be recorded are not located on company premises.

Examples:

o Home-based agents or enterprise workers.

o Mobile phones of knowledge workers when they conduct work related (and legally required recording) calls. e.g. insurance agents, brokers, physicians.

Use Case 8: Geographically distributed or centralized recording.

Enterprises such as banks, insurance agencies, and retail stores may have many locations, possibly up to thousands of small sites. Frequently only phones and network infrastructure are installed in branches, without local recording services. In cases where calls inside or between branches must be recorded, a centralized recording system in data centers together with telephony infrastructure (e.g. PBX) may be deployed.

Use Case 9: Record complex call scenarios.

The following is an example of a scenario where one call that is recorded must be associated with a related call that also must be recorded.

o A Customer is in a conversation with a Customer Service Agent.

o Agent puts Customer on hold in order to consult with a Supervisor.

o Agent enters into a conversation with Supervisor.

o Agent disconnects from Supervisor, then reconnects with Customer.

o The Supervisor call must be associated with the original customer call.

Use case 10: High availability and continuous recording.

Specific deployment scenarios present different requirements for system availability, error handling, etc. including:

o An SRS must always be available at call setup time.

o No loss of media recording, including during failure of an SRS.

o The Communication Session must be terminated (or suitable notification given to parties) in the event of a recording failure.

Use Case 11: Record multi-channel, multi-media session.

Some applications require the recording of more than one media stream, possibly of different types. Media are synchronized, either at storage or at playback.

Speech analytics technologies (e.g. word spotting, emotion detection, speaker identification) may require speaker-separated recordings for optimum performance.

Multi-modal Contact Centers may include audio, video, IM or other interaction modalities.

In trading floors environments, in order to minimize storage and recording system resources, it may be preferable to mix multiple concurrent calls (Communication Sessions) on different handsets/ speakers on the same turret into single recording session.

Use Case 12: Real-time media processing.

It must be possible for an SRS to support real-time media processing, such as speech analytics of trading floor interactions. Real-time analytics may be employed for automatic intervention (stopping interaction or alerting) if for example, a trader is not following regulations.

Speaker separation is required in order to reliably detect who is saying specific phrases.

5. Requirements

The following are requirements for SIP-based Media Recording:

o REQ-001 The mechanism MUST provide a means for using the SIP protocol for establishing, maintaining and terminating Recording Sessions between a Session Recording Client and a Session Recording Server.

o REQ-002 The mechanism MUST support the ability to record all CSs in their entirety.

o REQ-003 The mechanism MUST support the ability to record selected CSs in their entirety, according to policy.

o REQ-004 The mechanism MUST support the ability to record selected parts of selected CSs.

o REQ-005 The mechanism MUST support the ability to record a CS without loss of media of RS (for example, clipping media at the beginning of the CS) due to RS recording preparation and also, without impacting the quality or timing of the CS (for example, delaying the start of the CS while preparation for recording session). See Use Case 4 in Section 4 for more details.

o REQ-006 The mechanism MUST support the recording of IVR sessions.

o REQ-007 The mechanism MUST support the recording of RTP media types voice, DTMF (as defined by [RFC4733]), video, and text (as defined by

[RFC4103]).

o REQ-008 The mechanism MUST support the ability for an SRC to deliver mixed audio streams from multiple Communication Sessions to an SRS.

Note: A mixed audio stream is where several related Communication Sessions are carried in a single Recording Session. A mixed media stream is typically produced by a mixer function. The RS MAY be informed about the composition of the mixed streams through session metadata.

o REQ-009: The mechanism MUST support the ability for an SRC to deliver mixed audio streams from different parties of a given Communication Session to an SRS.

o REQ-010 The mechanism MUST support the ability to deliver to the SRS multiple media streams for a given CS.

o REQ-011 The mechanism MUST support the ability to pause and resume the transmission and collection of RS media.

o REQ-012 The mechanism MUST include a means for providing the SRS with metadata describing CSs that are being recorded, including the media being used and the identifiers of parties involved.

o REQ-013 The mechanism MUST include a means for the SRS to be able to correlate RS media with CS participant media.

o REQ-014 Metadata format must be agnostic of the transport protocol.

o REQ-015: The mechanism MUST support a means to stop the recording.

o REQ-016: The mechanism MUST support a means for a recording-aware UA involved in a CS to request at session establishment time that the CS should be recorded or should not be recorded, the honoring of such a request being dependent on policy.

o REQ-017: The mechanism MUST support a means for a recording-aware UA involved in a CS to request during a session that the recording of the CS should be started, paused, resumed or stopped, the honoring of such a request being dependent on policy. Such recording-aware UA MUST be notified about outcome of such requests.

o REQ-018 The mechanism MUST NOT prevent the application of tones or announcements during recording or at the start of a CS to support notification to participants that the call is being recorded or may be recorded.

o REQ-019 The mechanism MUST provide a means of indicating to recording-aware UAs whether recording is taking place, for appropriate rendering at the user interface.

o REQ-020 The mechanism MUST provide a way for metadata to be conveyed to the SRS incrementally during the CS.

o REQ-021 The mechanism MUST NOT prevent high availability deployments.

o REQ-022 The mechanism MUST provide means for facilitating synchronization of the recorded media streams and metadata.

o REQ-023 The mechanism MUST provide means for facilitating synchronization among the recorded media streams.

o REQ-024 The mechanism MUST provide means to relate recording and recording controls such as start/stop/pause/resume to the wall clock time.

o REQ-025 The mechanism MUST provide means for an SRS to authenticate the SRC on RS initiation.

o REQ-026 The mechanism MUST provide means for an SRC to authenticate the SRS on RS initiation.

o REQ-027 The mechanism MUST include a means for ensuring that the integrity of the metadata sent from SRC to SRS is an accurate representation of the original CS metadata.

o REQ-028 The mechanism MUST include a means for ensuring that the integrity of the media sent from SRC to SRS is an accurate representation of the original CS media.

o REQ-029 The mechanism MUST include a means for ensuring the confidentiality of the Metadata sent from SRC to SRS.

o REQ-030 The mechanism MUST provide a means to support RS confidentiality.

o REQ-031 The mechanism MUST support the ability to deliver to the SRS multiple media streams of the same media type (e.g. audio, video). For example in the case of delivering unmixed audio for each participant in the CS.

6. Privacy Considerations

Respecting the privacy rights and wishes of users engaged in a call is of paramount importance. In many jurisdictions participants have a right to know that the session is being recorded or might be recorded, and have a right to opt out, either by terminating the call or by demanding that the call not be recorded. Therefore this document contains requirements for being able to notify users that a call is being recorded and for users to be able to request that a call not be recorded. Use cases where users participating in a call are not informed that the call is or might be recorded are outside the scope of this document. In particular, lawful intercept is outside the scope of this document.

Requirements for participant notification of recording vary widely by jurisdiction. In a given deployment, not all users will be authorized to stop the recording of a CS (although any user can terminate its participation in a CS). Typically users within the domain that is carrying out the recording will be subject to policies of that domain concerning whether CSs are recorded. For example, in a call centre, agents will be subject to policies of the call centre and may or may not have the right to prevent the recording of a CS or part of a CS. Users calling into the call centre, on the other hand, will typically have to ask the agent not to record the CS. If the agent is unable to prevent recording, or if the caller does not trust the agent, the only option generally is to terminate the CS.

Privacy considerations also extend to what happens to a recording once it has been created. Typical issues are who can access the recording (e.g., receive a copy of the recording, view the metadata, play back the media, etc.), for what purpose the recording can be used (e.g., for training purposes, for quality control purposes, etc.) and for how long the recording is to be retained before deletion. These are typically policies of the domain that makes the recording, rather than policies of individual users involved in a recorded CS, whether those users be in the same domain or in a different domain. Taking the call centre example again, agents might be made aware of call centre policy regarding retention and use of recordings as part of their employment contract, and callers from outside the call centre might be given some information about policy when notified that a CS will be recorded (e.g., through an announcement that says that calls may be recorded for quality purposes).

This document does not specify any requirements for a user engaged in a CS to be able to dictate policy for what happens to a recording, or for such information to be conveyed from an SRC to an SRS. It is assumed that the SRS has access to policy applicable to its environment and can ensure that recordings are stored and used in accordance with that policy.

7. Security Considerations

Session recording has substantial security implications, for the SIP UA’s being recorded, the SRC, and the SRS.

For the SIP UA’s involved in the Communication Session, the requirements in this draft enable the UA to identify that a Communication Session is being recorded and for the UA to request that a given Communication Session is not subject to recording.

Since humans don’t typically look at or know about protocol signaling such as SIP, and indeed the SIP session might have originated through a PSTN Gateway without any ability to pass on in-signaling indications of recording, users can be notified of recording in the media itself through voice announcements, a visual indicator on the endpoint, or other means.

With regards to security implications of the protocol(s), clearly there is a need for authentication, authorization and eavesdropping protection for the solution. The SRC needs to know the SRS it is communicating with is legitimate, and vice-versa, even if they are in different domains. Both the signaling and media for the Recording Session need the ability to be authenticated and protected from eavesdropping. Requirements are detailed in the requirements section.

Communication Sessions and Recording Sessions can require different security levels both for signaling and media, depending on deployment configurations. For some environments, for example, SRS and SRC will be collocated in a secure network region and therefore the RS will not require the same protection level as a CS that extends over a public network, for example. For other environments, the SRS can be located in a public cloud, for example, and the RS will require a higher protection level than the CS. For these reasons, there is not a direct relationship between the security level of Communication Sessions and the security level of Recording Sessions.

A malicious or corrupt SRC can tamper with media and metadata relating to a CS before sending to an SRS. Also CS media and signaling can be tampered with in the network prior to reaching an SRC, unless proper means are provided to ensure integrity protection during transmission on the CS. Means for ensuring the correctness of media and metadata emitted by an SRC are outside the scope of this work. Other organizational and technical controls will need to be used to prevent tampering.

8. IANA Considerations

This document has no IANA actions.

9. Acknowledgements

Thanks to Dan Wing, Alan Johnson, Vijay Gurbani, Cullen Jennings, Hadriel Kaplan, Henry Lum, Dave Smith, Martin Palmer, Alissa Cooper, Deepanshu Gautam, Paul Kyzivat, Parthasarathi R, Ram Mohan R, and Charles Eckel for their significant contributions and assistance with this document and Working Group, and to all the members of the DISPATCH WG and SIPREC WG mailing lists for providing valuable input to this work.

10. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[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.

[RFC4103] Hellstrom, G. and P. Jones, "RTP Payload for Text Conversation", RFC 4103, June 2005.

[RFC4733] Schulzrinne, H. and T. Taylor, "RTP Payload for DTMF Digits, Telephony Tones, and Telephony Signals", RFC 4733, December 2006.

[RFC4949] Shirey, R., "Internet Security Glossary, Version 2", RFC 4949, August 2007.

Authors' Addresses

Ken Rehor (editor)
Cisco Systems
170 West Tasman Dr.
Mail Stop SJC30/2/
San Jose, CA 95134
USA

Email: krehor@cisco.com

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

Email: leon.portman@nice.com

Andrew Hutton
Siemens Enterprise Communications

Email: andrew.hutton@siemens-enterprise.com
URI: http://www.siemens-enterprise.com

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

Email: rajnish.jain@ipc.com

SIPREC L. Portman, Ed.
Internet-Draft NICE Systems
Intended status: Informational H. Lum, Ed.
Expires: January 9, 2012 Genesys, Alcatel-Lucent
A. Johnston
Avaya
A. Hutton
Siemens Enterprise
Communications
July 8, 2011

Session Recording Protocol
draft-portman-siprec-protocol-05

Abstract

The Session Recording Protocol is used for establishing recording session and reporting of the metadata of the communication session.

This document specifies the Session Recording Protocol. The protocol is used between Session Recording Client (SRC) and Session Recording Server (SRS).

Table of Contents

1. Introduction

Communication Session (CS) recording requires establishment of the recording session between communication system and recording system. In order to allow access to such recordings, the metadata about the CS shall be sent from the SRC to the SRS.

The SIP-based Media Recording Requirements [I-D.ietf-siprec-req] list a set of requirements that need to be met by session recording protocols. The Session Recording Protocol, which is specified in this document, meets these requirements.

The Session Recording Protocol uses SIP as the protocol for session establishment with special attention to reducing size of the required SIP messages. In addition, it is designed for future extendability and protocol version management to ensure backward compatability.

The remainder of this document is organized as follows: Section 2 defines the terminology used throughout this document, Section 3 discusses the scope of the Session Recording Protocol, Section 4 provides a non-normative overview of recording operations, Section 5 provides normative description of SIP extensions for the Recording Session, Section 6 provides normative description of SIP extensions for recording-aware user agents.

2. Definitions

The core defintions are taken from the requirements document [I-D.ietf-siprec-req].

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 a logical function that typically archives media for extended durations of time and provides interfaces for search and retrieval of the archived media. An SRS is typically implemented as a multi-port device that is capable of receiving media from several sources simultaneously. An SRS is typically also the sink of the recorded 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 itself or another entity working on its behalf (such as a SIP Application Server) may act as the source of the recording metadata.

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

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

Metadata Information that describes recorded media and the CS to which they relate.

Session Recording Protocol The set of SIP extensions that supports recording of Communication Sessions.

Pause during a Communication Session: The action of temporarily discontinuing the transmission and collection of RS media.

Resume during a Communication Session: The action of recommencing the transmission and collection of RS media.

Figure 1 shows the relationship between the defintions.

```
     +-------------+                                      +-----------+
     |             |        Communication Session         |           |
     |     A       |<------------------------------------>|     B     |
     |             |                                      |           |
     +-------------+                                      +-----------+
     ..................................................................
     .                             Session                            .
     .                            Recording                           .
     .                             Client                             .
     ..................................................................
                                      |
                                      | Recording
                                      | Session
                                      |
                                      v
                               +------------+
                               |   Session  |
                               |  Recording |
                               |   Server   |
                               +------------+
```

Figure 1: Relationship between CS, SRC, SRS, and RS

3. Scope

The scope of the Session Recording Protocol includes the establishment of the recording sessions and the reporting of the metadata. The following items, which is not an exhaustive list, do not represent the protcol itself and are considered out of the scope of the Session Recording Protocol:

- o Recording policies that determine whether the CS should be recorded
- o Retention policies that determine how long a recording is stored
- o Searching and accessing the recorded media and metadata
- o Delivering recording session metadata through non-SIP mechanism

4. Overview of operations

This section is informative and provides a description of recording operations.

As mentioned in the architecture document [I-D.ietf-siprec-architecture], there are a couple of types of call flows based on the location of the Session Recording Client. The following sample call flows provide a quick overview of the operations between the SRC and the SRS.

4.1. Delivering recorded media

When the SRC is deployed as a B2BUA, the SRC can route call requests from UA(A) to UA(B). As a SIP B2BUA, the SRC has access to the media path between the user agents. When the SRC is aware that it should be recording the conversation, the SRC may bridge the media between UA(A) and UA(B). The SRC then establishes the Recording Session with the SRS and sends replicated media towards the SRS.

An endpoint can also be acting as the SRC, and the endpoint itself will be establishing the Recording Session to the SRS. Since the endpoint has access to the media in the Communication Session, the endpoint can send replicated media towards the SRS.

The following is a sample call flow that shows the SRC establishing a recording session towards the SRS. The call flow is essentially identical when the SRC is a B2BUA or as the endpoint itself. Note that the SRC can choose when to establish the Recording Session independent of the Communication Session, even though the following call flow suggests that the Recording Session is established after the Communication Session is established.

```
   UA A           SRC                    UA B                   SRS
    |(1)CS INVITE  |                       |                      |
    |------------->|                       |                      |
    |              |(2)CS INVITE           |                      |
    |              |---------------------->|                      |
    |              |               (3)OK   |                      |
    |              |<----------------------|                      |
    |        (4)OK |                       |                      |
    |<-------------|                       |                      |
    |              |(5)RS INVITE with SDP  |                      |
    |              |--------------------------------------------->|
    |              |                       |       (6)OK with SDP |
    |              |<---------------------------------------------|
    |(7)CS RTP     |                       |                      |
    |=============>|======================>|                      |
    |<=============|<======================|                      |
    |              |(8)RS RTP              |                      |
    |              |=============================================>|
    |              |=============================================>|
    |(9)CS BYE     |                       |                      |
    |------------->|                       |                      |
    |              |(10)CS BYE             |                      |
    |              |---------------------->|                      |
    |              |(11)RS BYE             |                      |
    |              |--------------------------------------------->|
    |              |                       |                      |
```

Figure 2: Basic Recording Call flow

4.2. Conference focus as an SRC

A conference focus may also act as an SRC since it has access to all the media from each conference participant. In this example, a user agent may REFER the conference focus to the SRS, and the SRC may choose to mix media streams from all participants as a single media stream towards the SRS. In order to tell the conference focus to start a recording session to the SRS, the user agent can include the srs feature tag in the Refer-To header as per [RFC4508].

```
    UA A                  Focus                UA B              SRS
      |                   (SRC)                  |                  |
      |                     |                    |                  |
      |      (already in a conference)           |                  |
      |<==================>|<==================>|                  |
      |(1)REFER sip:Conf-ID Refer-To:<SRS>;srs  |                  |
      |------------------->|                                       |
      |(2)202 Accepted     |                                       |
      |<-------------------|                                       |
      |  (3)NOTIFY (Trying)|                                       |
      |<-------------------|                                       |
      |(4)200 OK           |                                       |
      |------------------->|                                       |
      |                    |(5)RS INVITE Contact:Conf-ID;isfocus   |
      |                    |-------------------------------------->|
      |                    |                             (6)200 OK |
      |                    |<--------------------------------------|
      |                    | (7)RTP (mixed or unmixed)             |
      |                    |======================================>|
      |     (8)NOTIFY (OK) |                                       |
      |<-------------------|                                       |
      |(9)200 OK           |                                       |
      |------------------->|                                       |
```

Figure 3: Recording call flow - SRC as a conference focus

4.3. Delivering recording metadata

Certain metadata, such as the attributes of the recorded media stream, are already included in the SDP of the recording session. This information is reused as part of the metadata. The SRC may provide an initial metadata snapshot about recorded media streams in the initial INVITE content in the recording session. Subsequent metadata updates can be represented as a stream of events in UPDATE or reINVITE requests sent by the SRC. These metadata updates are normally incremental updates to the initial metadata snapshot to optimize on the size of updates, however, the SRC may also decide to send a new metadata snapshot anytime.

The SRS also has the ability to sent a request to the SRC to request to receive a new metadata snapshot update when the SRS fails to understand the current stream of incremental updates for whatever reason (ie. SRS gets a syntax/semantic error in metadata update, the SRS crashes and restarts), and the SRS may attach a reason along with the snapshot request. This request allows both SRC and SRS to restart the states with a new metadata snapshot so that further metadata incremental updates will be based on the latest metadata snapshot. Similar to the metadata content, the metadata snapshot request is transported as content in UPDATE or INVITE sent by the SRS in the recording session.

```
     SRC                                                    SRS
      |                                                      |
      |(1) INVITE (metadata snapshot)                        |
      |----------------------------------------------------->|
      |                                           (2)200 OK  |
      |<-----------------------------------------------------|
      |(3) ACK                                               |
      |----------------------------------------------------->|
      |(4) RTP                                               |
      |=====================================================>|
      |(5) UPDATE (metadata update 1)                        |
      |----------------------------------------------------->|
      |                                          (6) 200 OK  |
      |<-----------------------------------------------------|
      |(7) UPDATE (metadata update 2)                        |
      |----------------------------------------------------->|
      |                                          (8) 200 OK  |
      |<-----------------------------------------------------|
      |              (9) UPDATE (metadata snapshot request)  |
      |<-----------------------------------------------------|
      |                   (10) 200 OK (metadata snapshot 2)  |
      |----------------------------------------------------->|
      |(11) UPDATE (metadata update 1 based on snapshot 2)   |
      |----------------------------------------------------->|
      |                                         (12) 200 OK  |
      |<-----------------------------------------------------|
```

Figure 4: Delivering metadata via SIP UPDATE

In some cases session metadata can be conveyed through non-SIP mechanism such as HTTP or JTAPI. These non-SIP mechanisms are considered out of the scope of the Session Recording Protocol, however, it is envisoned that a link with a URI can be provided in the recording session INVITE message so that the SRS can access the session metadata via the URI provided that the SRS supports the type of URI.

5. SIP Extensions for Recording Session

The following sections describe SIP extensions for the Recording Session.

The From header must contain the identity of the SRC. Participants information is not recorded in the From or To header; they are included in the metadata information.

Note that a recording session does not have to live within the scope of a single communication session. As outline in REQ-005 of [I-D.ietf-siprec-req], the recording session can be established in the absence of a communication session. In this case, the SRC must pre-allocate a recorded media stream and offer an SDP with at least one m= line to establish a persistent recording session. When the actual call arrives, the SRC can map recorded media stream to participant media and minimize media clipping.

Recorded media from multiple communication sessions may be handled in a single recording session. The SRC provides a reference of each recorded media stream to the metadata described in the next section.

5.1. Callee Capabilities Extensions for SIP Recording

This section discusses how the callee capabilities defined in [RFC3840] can be extended for SIP call recording.

SIP Callee Capabilities defines feature tags which are used to represent characteristics and capabilities of a UA. From RFC 3840:

> "Capability and characteristic information about a UA is carried as parameters of the Contact header field. These parameters can be used within REGISTER requests and responses, OPTIONS responses, and requests and responses that create dialogs (such as INVITE)."

Note that feature tags are also used in dialog modifying requests and responses such as re-INVITE and responses to a re-INVITE, and UPDATE. The 'isfocus' feature tag, defined in [RFC4579] is similar semantically to this case: it indicates that the UA is acting as a SIP conference focus, and is performing a specific action (mixing) on the resulting media stream. This information is available from OPTIONS queries, dialog package notifications, and the SIP registration event package.

We propose the definition of two new feature tags: 'src' and 'srs'.

5.1.1. src Feature Tag

The 'src' feature tag is used in Contact URIs by the Session Recording Client (SRC) related to recording sessions. A Session Recording Server uses the presence of this feature tag in dialog creating and modifying requests and responses to confirm that the dialog being created is for the purpose of a Recording Session. In addition, a registrar could discover that a UA is an SRC based on the presence of this feature tag in a registration. Other SIP Recording extensions and behaviors can be triggered by the presence of this feature tag.

Note that we could use a single feature tag, such as 'recording' used by either an SRC or SRS to identify that the session is a recording session. However, due to the differences in functionality and behavior between an SRC and SRS, using only one feature tag for both is not ideal. For instance, if a routing mistake resulted in a request from a SRC being routed back to another SRC, if only one feature tag were defined, they would not know right away about the error and could become confused. With separate feature tags, they would realize the error immediately and terminate the session. Also, call logs would clearly show the routing error.

5.1.2. srs Feature Tag

The 'srs' feature tag is used in Contact URIs by the Session Recording Server (SRS) related to recording sessions. A Session Recording Client uses the presence of this feature tag in dialog creating and modifying requests and responses to confirm that the dialog being created is for the purpose of a Recording Session (REQ-30). In addition, a registrar could discover that a UA is an SRS based on the presence of this feature tag in a registration. Other SIP Recording extensions and behaviors can be triggered by the presence of this feature tag.

To ensure a recording session is redirected to an SRS, an SRC can utilize the SIP Caller Preferences extensions, defined in [RFC3841]. The presence of a Accept-Contact: *;sip.srs allows a UA to request that the INVITE be routed to an SRS. Note that to be completely sure, the SRC would need to include a Require: prefs header field field in the request.

5.2. SDP handling

Following the SDP offer/answer model in [RFC3264], this section describes the conventions used in the recording session for SDP handling.

SRC must provide an SDP offer in the initial INVITE to the SRS. SRC can include one or more media streams to the SRS. The SRS must respond with the same number of media descriptors in the SDP body of the 200 OK.

The SRC should use a=sendonly attribute as the SRC does not expect to receive media from the SRS. As SRS only receives RTP streams from

SRC, the 200 OK response will normally contain SDP with a=recvonly attribute.

Since the SRC may send recorded media of different participants (or even mixed streams) to the SRS, the SDP must provide a label on each media stream in order to identify the recorded stream with the rest of the metadata. The a=label attribute [RFC4574] will be used to identify each recorded media stream, and the label name is mapped to the Media Stream Reference in the metadata in [I-D.ietf-siprec-metadata]. Note that a participant may have multiple streams (audio and video) and each stream is labeled separately.

```
v=0
o=SRS 0 0 IN IP4 172.22.3.8
s=SRS
c=IN IP4 172.22.3.8
t=0 0
m=audio 12241 RTP/AVP 0 4 8
a=sendonly
a=label:1
m=audio 12242 RTP/AVP 98
a=rtpmap:98 H264/90000
a=fmtp:98 ...
a=sendonly
a=label:2
m=audio 12243 RTP/AVP 0 4 8
a=sendonly
a=label:3
m=audio 12244 RTP/AVP 98
a=rtpmap:98 H264/90000
a=fmtp:98 ...
a=sendonly
a=label:4
```

Figure 6: Sample SDP with audio and video streams

To remove a recorded media stream from the recording session, send a reINVITE and set the port to zero in the m= line.

To add a recorded media stream, send a reINVITE and add a new m= line.

The SRS may respond with a=inactive attribute as part of the SDP in the 200 OK response when the SRS is not ready to receive recorded media. The SRS can send re-INVITE to update the SDP with a=recvonly when it is ready to receive media.

The following sequence diagram shows an example of SRS responds with SDP that contain a=inactive, and then later update media information update with re-INVITE.

```
     SRC                                                     SRS
      |                                                       |
      |(1) INVITE (SDP offer)                                 |
      |------------------------------------------------------>|
      |                         (2)200 OK with SDP inactive   |
      |<------------------------------------------------------|
      |(3) ACK                                                |
      |------------------------------------------------------>|
      |                      ...                              |
      |                      (4) re-INVITE with SDP recvonly  |
      |<------------------------------------------------------|
      |(5)200 OK with SDP sendonly                            |
      |------------------------------------------------------>|
      |                                             (6) ACK   |
      |<------------------------------------------------------|
      |(7) RTP                                                |
      |======================================================>|
      |                      ...                              |
      |(8) BYE                                                |
      |------------------------------------------------------>|
      |                                             (9) OK    |
      |<------------------------------------------------------|
```

Figure 7: SRS to offer with a=inactive

5.3. RTP handling

[This is a placeholder section to specify any protocol impacts or recommendations for RTP usage in the session recording protocol. The details are listed in [I-D.eckel-siprec-rtp-rec]]

5.4. Metadata

The format of the full metadata will be described as part of the mechanism in [I-D.ietf-siprec-metadata].

As mentioned in the previous section, the SDP of the recording session is the metadata for all recorded media streams. The label attribute contains a reference to the rest of the metadata information.

For all basic metadata information such as communication session, participants, call identifiers and direction, they can be included in the initial INVITE request sent by the SRC. Metadata can be included as content in the INVITE or UPDATE request. A new "disposition-type" of Content-Disposition is defined for this purpose and the value is "recording-session".

The following SIP example for RS establishment between SRC and SRS with metadata as content.

```
INVITE sip:97753210@10.240.3.10:5060 SIP/2.0
From: <sip:2000@10.226.240.3>;tag=35e195d2-947d-4585-946f-098392474
To: <sip:Recorder@10.240.3.10>
Call-ID: d253c800-b0d1ea39-4a7dd-3f0e20a@10.226.240.3
CSeq: 101 INVITE
Date: Thu, 26 Nov 2009 02:38:49 GMT
Supported: timer
Supported: replaces
User-Agent: B2BUA
Max-Forwards: 70
Allow: INVITE,OPTIONS,INFO,BYE,CANCEL,ACK,PRACK,UPDATE,
  REFER,SUBSCRIBE,NOTIFY,PUBLISH
Allow-Events: presence,kpml
    Min-SE: 90
Contact: <sip:2000@10.226.240.3:5060;transport=tcp>;isfocus;src
Via: SIP/2.0/TCP 10.226.240.3:5060;branch=z9hG4bKdf6b622b648d9
Session-Expires: 1800
Content-Type: multipart/mixed;boundary=foobar
Content-Length: [length]

--foobar
Content-Type: application/sdp

v=0
o=SRS 0 0 IN IP4 10.226.240.3
c=IN IP4 10.226.240.3
t=0 0
m=audio 12241 RTP/AVP 0 4 8
a=sendonly
a=label:1

--foobar
Content-Type: application/rs-metadata
Content-Disposition: recording-session

[metadata content]
```

Figure 8: Sample INVITE request for the recording session

Further updates to recording metadata can be deliverd as a sequence events reported in SIP UPDATE or reINVITE requests and the SRS must receive the sequence of events in order. Since there can only be a single INVITE or UPDATE transaction happening at a time within a SIP dialog, using sequence number CSeq in the dialog can be a reliable way for the SRS to identify the receipt of the next metadata update.

At any time during Recording Session, the SRC may send a new metadata snapshot in SIP UPDATE or reINVITE request. All subsequent metadata updates will be based on the new metadata snapshot.

5.5. Requesting for metadata snapshot

The SRS may send a request for metadata snapshot any time after the Recording Session has been established. Typically, the SRS sends such as request in the case where the SRS is failing to process further metadata incremental updates. Failure scenarios can include failure to parse metadata information (syntax error), failure to match metadata information with the current metadata snapshot (semantic error), or failure at the SRS.

Similar to delivering metadata, the SRS sends the metadata snapshot request as content in UPDATE or INVITE requests or responses. The same disposition type "recording-session" is used to note that the content represents content sent by the SRS. The format of the content is application/rs-metadata-request, and the body format is chosen to be a simple text-based format with header and values. The following shows an example:

```
SRS-Status: SRS failure
```

The SRS MUST include the reason why a metadata snapshot request is being made to the SRC in the SRS-Status header. This header is free form text to allow the SRS to provide a descriptive reason. The body format also allows additional extension headers to be included by the SRS in the snapshot request to convey additional information to the SRC.

When the SRC receives the request for a metadata snapshot, the SRC may provide the metadata snapshot in the response or as a separate INVITE/UPDATE transaction. All subsequent metadata updates sent by the SRC MUST be based on the new metadata snapshot.

5.5.1. Formal Syntax

The formal syntax for the application/rs-metadata-request MIME is described below using the augmented Backus-Naur Form (BNF) as described in [RFC2234].

```
snapshot-request = srs-status-line CRLF [ *opt-srs-headers ]

srs-status-line = "SRS-Status" HCOLON srs-status

srs-status = [TEXT-UTF8-TRIM]

opt-srs-headers = CRLF 1*(extension-header CRLF)
```

5.6. Recording Pause and Resume

To temporarily discontinue streaming and collection of recorded media from the SRC to the SRS, the SRC must send a reINVITE and set a=inactive for each recorded media stream to be paused.

To resume streaming and collection of recorded media, the SRC must send a reINVITE and set a=sendonly for each recorded media stream to resume.

Note that when a media stream in the CS is muted/unmuted, this information may be conveyed in the metadata by the SRC. The SRC should not modify the recorded media stream with a=inactive for mute since this operation is reserved for pausing the RS media.

6. SIP Extensions for Recording-aware User Agents

The following sections describe SIP extensions for recording-aware UA.

6.1. Providing recording indication

While there are existing mechanisms for providing an indication that a CS is being recorded, these mechanisms are usually delivered on the CS media streams such as playing an in-band tone or an announcement to the participants. A new SDP attribute is introduced to allow a recording-aware UA to render recording indication at the user interface.

The ’record’ SDP attribute appears at the media level, and may appear in either SDP offer or answer. The recording indication applies to the specified media stream only, for example, the audio portion of the call may be recorded in a audio/video call. The following is the

ABNF of the 'record' attribute:

```
record-attr = "a=record:" indication

indication = "on" / "off" / "paused"
```

on Recording is in progress.

off No recording is in progress.

paused Recording is in progress by media is paused.

The recording attribute is a declaration by the endpoints in the session to indicate whether recording is taking place. For example, if a UA (A) is initiating a call to UA (B) and UA (A) is also an SRC that is performing the recording, then UA (A) provides the recording indication in the SDP offer with a=record:on. When UA (B) receives the SDP offer, UA (B) will see that recording is happening on the other endpoint of this session. If UA (B) does not wish to perform recording itself, UA (B) provides the recording indication as a=record:off in the SDP answer.

Whenever the recording indication needs to change, such as termination of recording, then the UA must initiate a reINVITE to update the SDP attribute to a=record:off. The following call flow shows an example of the offer/answer with the recording indication attribute.

```
    UA A                                                   UA B
    (SRC)                                                   |
      |                                                     |
      |                [SRC recording starts]               |
      |(1) INVITE (SDP offer + a=record:on)                 |
      |---------------------------------------------------->|
      |                  200 OK (SDP answer + a=record:off) |
      |<----------------------------------------------------|
      |(3) ACK                                              |
      |---------------------------------------------------->|
      |(4) RTP                                              |
      |<===================================================>|
      |               [SRC stops recording]                 |
      |(5) re-INVITE (SDP + a=record:off)                   |
      |---------------------------------------------------->|
      |                      (6) 200 OK (SDP + a=record:off)|
      |<----------------------------------------------------|
      |                                             (6) ACK |
      |---------------------------------------------------->|
```

Figure 9: Recording indication example

If a call is traversed through one or more SIP B2BUA, and it happens that there are more than one SRC in the call path, the recording indication attribute does not provide any hint as to which SRC is performing the recording, meaning the endpoint only knows that the call is being recorded. This attribute is also not used as an indication to negotiate which SRC in the call path will perform recording if there are multiple SRCs in the call path.

6.2. Recording awareness

A recording-aware UA may indicate that it can accept reporting of recording indication in media level SDP provided in the previous section. A new option tag "record-aware" is introduced to indicate such awareness.

A UA that has indicated recording awareness by including the record-aware option tag in a transmitted Supported header field MUST provide at its user interface an indication whether recording is on or off for a given medium based on the most recently received a=record SDP attribute for that medium.

Some user agents that are automatons (eg. IVR, media server, PSTN gateway) may not have an user interface to render recording indication. When such user agent indicates recording awareness, these UA may render recording indication through other means, such as passing an inband tone on the PSTN gateway, putting the recording indication in a log file, or raising an application event in a VoiceXML dialog. These user agents may also choose not to indicate recording awareness, thereby relying on whatever mechansim an SRC chooses to indicate recording, such as playing a tone inband.

When a UA has not indicated that it is recording aware, an SRC must provide recording indications, where SRC is required to do so based on policies, through other means such as playing a tone inband.

6.3. Recording preference

A recording-aware UA involved in a CS may request the CS to be recorded or not recorded. This indication of recording preference may be sent at session establishment time or during the session.

A new SDP attribute "recordpref" is introduced. The SDP attribute appears at the media level and can only appear in an SDP offer. The recording indication applies to the specified media stream only. The following is the ABNF of the recordpref attribute:

```
recordpref-attr = "a=recordpref:" pref

pref = "on" / "off" / "pause" / "nopreference"
```

on Request for recording if it has not already been started. If the recording is currently paused, request to resume recording.

off Request for no recording. If recording has already been started, then this preference indicates a request to stop recording.

pause Request to pause recording if recording is currently in progress.

nopreference To indicate that the UA has no preference on recording. While the absence of this attribute indirectly implies the lack of preference, using this value allows the UA to explicitly state no preference to being recorded.

7. IANA Considerations

7.1. New Content-Disposition Parameter Registrations

This document registers a new "disposition-type" value in Content-Disposition header: recording-session.

recording-session the body describes the metadata information about the recording session

7.2. Media Type Registration

7.2.1. Registration of MIME Type application/rs-metadata

This document registers the application/rs-metadata MIME media type in order to describe the recording session metadata. This media type is defined by the following information:

Media type name: application

Media subtype name: rs-metadata

Required parameters: none

Options parameters: none

7.2.2. Registration of MIME Type application/rs-metadata-request

This document registers the application/rs-metadata-request MIME media type in order to describe a recording session metadata snapshot request. This media type is defined by the following information:

Media type name: application

Media subtype name: rs-metadata-request

Required parameters: none

Options parameters: none

7.3. Registration of record-aware Option Tag

This document registers the "record-aware" option tag.

Name: record-aware

Description: This option tag is to indicate the ability for the user agent to receive recording indicators in media level SDP. When present in a Supported header, it indicates that the UA can receive recording indicators in media level SDP.

7.4. SDP Attributes

This document registers the following new SDP attributes.

7.4.1. 'record' SDP Attribute

Attribute name: record

Long form attribute name: Recording Indication

Type of attribute: media level

Subject to charset: no

This attribute provides the recording indication for the session or media stream.

Allowed attribute values: on, off, paused

7.4.2. 'recordpref' SDP Attribute

Attribute name: recordpref

Long form attribute name: Recording Preference

Type of attribute: media level

Subject to charset: no

This attribute provides the recording indication for the session or media stream.

Allowed attribute values: on, off, pause, nopreference

8. Security Considerations

The recording session is fundamentally a standard SIP dialog [RFC3261], therefore, the recording session can reuse any of the existing SIP security mechanism available for securing the recorded media as well as metadata.

8.1. Authentication and Authorization

The recording session reuses the SIP mechanism to challenge requests that is based on HTTP authentication. The mechanism relies on 401 and 407 SIP responses as well as other SIP header fields for carrying challenges and credentials.

The SRS may have its own set of recording policies to authorize recording requests from the SRC. The use of recording policies is outside the scope of the Session Recording Protocol.

9. References

9.1. Normative References

[I-D.ietf-siprec-metadata]
R, R., R, P., and P. Kyzivat, "Session Initiation Protocol (SIP) Recording Metadata", draft-ietf-siprec-metadata-03 (work in progress), July 2011.

[I-D.ietf-siprec-req]
Rehor, K., Portman, L., Hutton, A., and R. Jain, "Use Cases and Requirements for SIP-based Media Recording (SIPREC)", draft-ietf-siprec-req-12 (work in progress), June 2011.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", RFC 2234, November 1997.

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

[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.

[RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, June 2002.

[RFC3840] Rosenberg, J., Schulzrinne, H., and P. Kyzivat, "Indicating User Agent Capabilities in the Session Initiation Protocol (SIP)", RFC 3840, August 2004.

[RFC3841] Rosenberg, J., Schulzrinne, H., and P. Kyzivat, "Caller Preferences for the Session Initiation Protocol (SIP)", RFC 3841, August 2004.

[RFC4574] Levin, O. and G. Camarillo, "The Session Description Protocol (SDP) Label Attribute", RFC 4574, August 2006.

9.2. Informative References

[I-D.eckel-siprec-rtp-rec]
Eckel, C., "Real-time Transport Protocol (RTP) Recommendations for SIPREC", draft-eckel-siprec-rtp-rec-01 (work in progress), June 2011.

[I-D.ietf-siprec-architecture]
Hutton, A., Portman, L., Jain, R., and K. Rehor, "An Architecture for Media Recording using the Session Initiation Protocol", draft-ietf-siprec-architecture-02 (work in progress), April 2011.

[RFC4508] Levin, O. and A. Johnston, "Conveying Feature Tags with the Session Initiation Protocol (SIP) REFER Method", RFC 4508, May 2006.

[RFC4579] Johnston, A. and O. Levin, "Session Initiation Protocol (SIP) Call Control - Conferencing for User Agents", BCP 119, RFC 4579, August 2006.

Authors' Addresses

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

Email: leon.portman@nice.com

Henry Lum (editor)
Genesys, Alcatel-Lucent
1380 Rodick Road, Suite 200
Markham, Ontario L3R4G5
Canada

Email: henry.lum@genesyslab.com

Alan Johnston
Avaya
St. Louis, MO 63124

Email: alan.b.johnston@gmail.com

Andrew Hutton
Siemens Enterprise Communications

Email: andrew.hutton@siemens-enterprise.com

Session Initiation Protocol (SIP) Recording Metadata
draft-ram-siprec-metadata-04

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 the metadata model as viewed by Session Recording Server(SRS).

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 focuses on the Recording metadata which describes the communication session. The document describes a metadata model as viewed by Session Recording Server, the requirements for which are described in [I-D.ietf-siprec-req] and the architecture for which is described in [I-D.ietf-siprec-architecture].

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 [RFC2119]. This document only uses these key words when referencing normative statements in existing RFCs."

3. Metadata Model

Metadata is the data that describes the communication session. Below diagram shows a model for Metadata as viewed by Session Recording Server (SRS).

```
  +-------------------------------+        1
  | Recording Session (RS)        |---------------+
  +-------------------------------+               |
             | 1..*                               |
             |                                    |
             | 0..*                               |
  +-------------------------------+               |
  |   Communication Session (CS)  |        1      |
  |      Group                    |---------------|
  +-------------------------------+               |
             | 1                                  |
             |                                    |
             | 1..*                               |
  +-------------------------------+               |
  |   Communication Session (CS)  |        1      |
  |                               |---------------|
  +-------------------------------+               |       +-------------+
        | 0..*                 |1..*              |       |             |
        |                      |                  |  0..* |Extension    |
        | 2..*                 |0..*              |-------| Data        |
  +-------------+ receives +----------------+     |       |             |
  | Participant |----------|  Media Streams |     |       +-------------+
  |             |0..*  0..*|                |     |
  |             |          |                |     |
  |             |          |                |     |
  |             |  sends   |                |     |
  |             |----------|                |     |
  |             |1.*   0..*|                |     |
  +-------------+          +----------------+     |
         |                          |             |
         |1                         |1            |
         |                          |             |
         +----------------------------------------+
```

The mechanism MUST provide a means to convey every attribute mentioned in the metamodel. Session Recording Client (SRC) MAY initiate the Recording Session. It should be noted that the Recording Session is a completely independent from the Communication Session that is being recorded at both the SIP dialog level and at the session level. The metadata MUST be conveyed from SRC to SRS. The metadata MAY be conveyed in Recording Session Dialog.

Note that the metadata model captures changes that occur over the duration of the recording session. For example, if the call is transferred from one participant to another, then the SRC SHALL convey a change of participant and the properties of the new media stream to the SRS.

Some of the data in the model may not be conveyed explicitly from the SRC to the SRS, if it can be obtained contextually by the SRS. For instance, the timing of changes may not explicitly conveyed from the SRC to the SRC, because the mechanism (yet to be defined) which conveys the metadata may implicitly provide the timing. (E.g. the time a change occurred by be assumed to be the same as the time when notification of the change is received by the SRS.)

4. Recording Metadata elements

This section describes the different elements and its attributes of the metadata model shown above. This section also describes in brief on how the different elements of metadata are associated.

4.1. Recording Session

```
   +--------------------------------+
   | Recording Session (RS)         |
   +--------------------------------+
   |  Recording RequestorID(SRC or  |            +------------------+
   |                   SRS)         |    1  0..* |                  |
   |                                |------------|Extension Data    |
   |  Recording Type (Selective     |            |                  |
   |  Persistant)                   |            +------------------+
   +--------------------------------+
              | 1..*
              |
              | 0..*
    Communication Session Group(CS Group)
```

A Recording Session element represents one instance of a Recording Session.

4.1.1. Attributes

A Recording Session element MAY have attributes like:
- o Recording requestor ID(which could be SRS or SRC).
- o Recording type - This attribute indicates whether the recording session is selective or persistent.

4.1.2. Associations

One instance of Recording Session SHALL have:

- o Zero or more instances of Communication Session Group.The allowance of zero instances is to accommodate persistent recording, where there may be none.
- o Each CS Group MUST be associated with one or more Recording Sessions [setup by the same SRC.]

4.2. Communication Session Group

```
    Recording Session (RS)
               | 1..*
               |
               | 0..*
   +--------------------------------+
   | Communication Session          |
   |     Group                      |
   +--------------------------------+
   |  Unique-ID                     |          +-----------------+
   |                                |   1  0..* |                 |
   |                                |-----------|Extension Data   |
   |                                |           |                 |
   +--------------------------------+           +-----------------+
               | 1
               |
               | 1..*
    Communication Session (CS)
```

A Communication Session Group provides association or linking of Communication Sessions.

4.2.1. Attributes

A CS Group MUST have a Unique-ID attribute. This Unique-ID is to group different CSs that are related. SRC (or MAY be SRS) MUST ensure the uniqueness of Unique-ID in case multiple SRC interacts with the same SRS. The mechanism by which SRC creates this unique-ID and ensures its uniqueness is outside the scope of SIPREC.

NOTE: Need more clarity/use cases on how the unique-ID SHALL be used

4.2.2. Associations

A communication Session Group SHALL be associated with RS and CS in the following manner:

- o There can be one or more Recording Session elements per Communication Session Group.
- o Each Communication Session Group MUST be associated with one or more RS [setup by the same SRC]
- o There MAY be one or more Communication Sessions per CS Group [e.g. Consult Transfer]
- o Each CS MUST be associated to one CS-Group

4.3. Communication Session

```
           Communication Session Group(CS Group)
              | 1
              |
              | 1..*
+-------------------------------+               +------------------+
|   Communication Session (CS)  |       1  0..* |                  |
|                               |---------------|Extension Data    |
+-------------------------------+               |                  |
|   CS Identifier               |               +------------------+
|   Call Termination Reason     |
|                               |
+-------------------------------+
   |                      |
   | 0..*                 |1..*
   |                      |
   | 2..*                 |0..*
Participant              Media Stream
```

A Communication Session block/element in the metadata model represents Communication Session and its properties needed as seen by SRC.

4.3.1. Attributes

A communication Session block SHALL have the following attributes:

- o Call Termination Reason - This represents the reason why a CS was terminated. The communication session MAY contain a Call Termination Reason. This MAY be derived from SIP Reason header of

CS.

- o CS Identifier - This attribute is used to uniquely identify a CS.

NOTE: Attributes like Retention (represent the value/duration for which Media streams of the CS needs to be retained), Force Deletion, Access Information e.t.c that are primarily related to policy will not be passed in metadata from SRC to SRS. However if there are implementations where SRC has enough information, this could be sent as Extension Data attached to CS

4.3.2. Associations

A Communication Session SHALL be associated to CS-Group,Participant and Media Stream. Cardinalities between CS and Participant allows:

- o CS to have atleast two or more participants
- o Participant may be associated with zero or more CS's (It is possible, though unlikely, that there are participants who are not part of any CS). An example of such a case is participants in a premixed media stream. The SRC may have knowledge of such Participants, yet not have any signaling relationship with them. This might arise if one participant in CS is a conf focus. Another use case is if one UA in CS works in 3pcc mode to acquire an MoH media stream, this might be reflected as unique source for media stream without having a reported signaling relationship to it.
- o The model also allows participants in CS that are not participants in the media. An example is the identity of a 3pcc controller that has initiated a CS to two or more participants of the CS. Another example is the identity of a conference focus. Of course a focus is probably in the media, but since it may only be there as a mixer, it may not report itself as a participant in any of the media streams.

Cardinalities between CS and Media Stream allows:

- o A CS to have zero or more Streams
- o A stream can be associated with 1 or more CS. An example is multicast MoH stream which might be associated with many CSs. Also if we were to consider a B2BUA to have a separate CS on each "side" then they might share a stream.(Though more likely this would be treated as a single CS.)

4.4. Participant

```
    Communication Session (CS)
               | 0..*
               |
               | 2..*
   +--------------------------------+
   |          Participant           |
   |                                |
   +--------------------------------+
   |  AoR list                      |            +------------------+
   |  Name                          |     1  0..* |                  |
   |  Participant Type              |------------|Extension Data    |
   |                                |            |                  |
   +--------------------------------+            +------------------+
            | 0..*       1..*|
   receives|                 |sends
            | 0..*       0..*|
             Media Stream
```

A Participant block has information about a device that is part of a CS and/or contributes/consumes media stream(s) belonging to a CS.

4.4.1. Attributes

Participant has attributes like:

- o AoR list - Has list of AoRs. An AoR MAY be SIP/SIPS/TEL URI. There MAY be cases where a participant can have more than one AoR [e.g. P-Asserted-ID which can have both SIP and TEL URIs]
- o Name - This attribute represents Participant name(SIP display name) or DN number (in case it is known)
- o Participant Type - This attribute can have values as "internal" or "external" or "don't know" (in cases where it is not possible to determine).

NOTE: Other attributes [like Participant Role] MAY be carried as part of extension data to Participant from SRC to SRS.

4.4.2. Associations

Cardinalities between participant and Media Stream allows:

- o Participant to receives zero or more media streams
- o Participant to send zero or more media streams. (Same participant provides multiple streams e.g. audio and video)

o Media stream to be received by zero or more participants. Its possible, though perhaps unlikely, that a stream is generated but sent only to the SRC and SRS, not to any participant. E.g. In conferencing where all participants are on hold and the SRC is collocated with the focus. Also a media stream may be received by multiple participants (e.g. Whisper calls, side conversations).
o Media stream to be sent by one or more participants (pre-mixed streams).

NOTE: Example of a case where a participant may receive Zero or more streams - a Supervisor may have side conversation with Agent, while Agent converses with customer.

4.5. Media Stream

```
                              Participant
                            | 0..*       1..*|
                    receives|                |sends
                            | 0..*       0..*|
                         +--------------------------+
                         |          Media Stream    |
                         |                          |
Communication 1..*  0..* +--------------------------+
  Session    ------------|  Start Time              |        +-----------+
                         |  End Time                |1  0..* |           |
                         |  Codec params            |--------|Extension  |
                         |  Media Stream Reference  |        | Data      |
                         +--------------------------+        +-----------+
```

A Media Stream block shall have properties of media as seen by SRC and sent to SRS. Different instances of Media Stream block would be created whenever there is a change in media (e.g. dir change like pause/resume and/or codec change and/or participant change.).

4.5.1. Attributes

A Media Stream block SHALL have the following attributes:

o Start Time - Represents Media Start time at SRC.
o End Time - Represents Media End time at SRC. This is an optional attribute and MAY be included after a stream ends
o Codec params - represents codec parameters of the CS media
o Media Stream Reference - In implementations this can reference to m-line

There may cases where SRC offered certain media types but SRS chooses to accept only a subset of them OR an SRC may not even offer a certain media type due it its restrictions to record. In such cases SRC MAY continue to send information about media streams that are not recorded to SRS in the metadata.

4.5.2. Associations

A Media Stream SHALL be associated with Participant and CS. The details of association with the Participant are described in the Participant block section. The details of association with CS is mentioned in the CS section.

4.6. Extension Data

A recording metadata object contains additional data not specified as part of siprec. This is intended to accommodate future standards track extensions, as well as vendor and user specific extensions. The mechanism MUST provide a means of unambiguously distinguishing such extension data.

5. Metadata Model Object Instances

This section describes the metadata model object instances for different use cases of SIPREC. For the sake of simplicity as the media streams sent by each of the participants is received by every other participant in these use cases, it is NOT shown in the object instance diagrams below.

5.1. Use case 1: Basic Call

Basic call between two Participants A and B. In this use case each participant sends one Media Stream. For the sake of simplicity "receives" lines are not shown in this instance diagram. Media Streams sent by each participant is received all other participants of that CS.

```
   +--------------------------------+
   | Recording Session (RS)         |
   +--------------------------------+
               |
               |
               |
   +--------------------------------+
   |   Communication Session (CS)   |
   |      Group(CSG)                |
   +--------------------------------+
   |   Unique-id1                   |
   +--------------------------------+
               |
               |
               |
     +-----------------+
     | Communication   |
     | Session (CS)    |
     +-----------------+
     |                 |
     +-----------------+
               |
               |-------------------+
               |        |          |
   +----------------+   | +----------------+
   | ParticipantA   |   | | ParticipantB   |
   |                |   | |                |
   +----------------+   | +----------------+
               |        |          |
     sends     |        |          |  sends
               |        |          |
   +----------------+   |   +----------------+
   |Media Stream A1|    |   |Media Stream B1|
   +----------------+   |   +----------------+
   |MediaStream Ref|    |   |MediaStream Ref|
   |codec  params  |---+---|codec params   |
   +----------------+       +----------------+
```

5.2. Use case 2: Basic Call with hold/resume

Basic call between two Participants A and B and with Participant A or B doing a Hold/Resume. In this use case each participant sends one Media Stream. After Hold/Resume the properties of Media MAY change. For the sake of simplicity "receives" lines are not shown in this instance diagram. Media Streams sent by each participant is received all other participants of that CS.

```
   +-------------------------------+
   | Recording Session (RS)        |
   +-------------------------------+
              |
              |
              |
   +-------------------------------+
   |   Communication Session (CS)  |
   |      Group(CSG)               |
   +-------------------------------+
   |   Unique-id1                  |
   +-------------------------------+
              |
              |
              |
     +----------------+
     | Communication  |
   +-| Session (CS)   |----------------------------------------------+
   | +----------------+                                              |
   | |                |                                              |
   | +----------------+                                              |
   |            |                                                    |
   |            |-------------------+                                |
   |            |                   |                                |
   | +---------------+     +---------------+                         |
   | | ParticipantA  |     | ParticipantB  |-----------+             |
   | |               |--+  |               |           |             |
   | +---------------+  |  +---------------+           |sends(After  |
   |       |       |    |       |         |            |     Resume) |
   |       |       |    |       |         |         +--------------+ |
   | sends |       |     +--+   |  sends  |         |MediaStream B3| |
   |       |       -----+   |   |          +-----+  +--------------+ |
   | +---------------+  |   |  +---------------+ |  |MediaStreamRef|-|
   | |Media Stream A1|  |   |  |Media Stream B1| |  |Codec Params  | |
   | +---------------+  |   |  +---------------+ |  |              | |
   +-|MediaStreamref |  |   |  |MediaStreamRef | |  +--------------+ |
     |codec params   |  |   |  |codec params   |-|-------------------|
     +---------------+  |   |  +---------------+ |                   |
                        |   |                    |                   |
           +------------+   |sends               |sends (hold)       |
           | sends          |(Resume)            |                   |
           | (hold)         +-------+             +-------+          |
           |                        |                     |          |
     +---------------+       +---------------+   +--------------+    |
     |Media Stream A2|       |Media Stream A3|   |MediaStream B2|    |
     +---------------+       +---------------+   |              |    |
     |MediaStreamref |       |MediaStreamRef |   +--------------+    |
     |codec params   |       |codec params   |   |Codec Params  |    |
```

```
     +---------------+       +---------------+   |MediaStreamRef|    |
              |                     |            |              |    |
              |                     |            +--------------+    |
              |                     |                  |             |
              +------------------------------------------------------+
```

NOTE: Need discssions on how to represent Hold/Resume from SRC to SRS and Pause/Resume from SRS to SRC.

5.3. Use case 3: Basic call with Transfer

Basic call between two Participants A and B and with Participant A transfer(consult transfer) to Participant C. In this use case each participant sends one Media Stream. After transfer the properties of Participant A Media MAY change. For the sake of simplicity "receives" lines are not shown in this instance diagram. Media Streams sent by each participant is received all other participants of that CS.

```
   +-------------------------------+
   | Recording Session (RS)        |
   +-------------------------------+
              |
              |
              |
   +-------------------------------+
   |   Communication Session (CS)  |
   |      Group(CSG)               |
   +-------------------------------+
   |   Unique-id1                  |
   +-------------------------------+
              |
              |-----------------
              |                 |
   +----------------+     +----------------+
   | Communication  |     | Communication  |
   | Session (CS)1  |     | Session (CS)2  |
   +----------------+     +----------------+-----------+
   |                |     |                |           |
   +----------------+     +----------------+           |
              |                                        |
              |-------------------+                    |
              |        |          |                    |
   +---------------+   | +---------------+             |
   | ParticipantA  |   | | ParticipantB  |             |
   |               |   | |               |             |
```

```
   +---------------+   | +---------------+             |
              |        |         |                     |
     sends    |        |         |  sends              |
              |        |         |                     |
   +---------------+   |   +---------------+           |
   |Media Stream A1|   |   |Media Stream B1|           |
   +---------------+   |   +---------------+           |
   |               |   |   |               |           |
   |codec params   |   |   | Media Stream  |           |
   | Media Stream  |---+---|  Ref          |           |
   | Ref           |       |codec params   |           |
   +---------------+       +---------------+           |
                                                       |
                                                       |
                          +----------------------------|
                          |                            |
         +--------------------------------+            |
         |                                |            |
   +---------------+                +---------------+  |
   | Participant A |                | Participant C |  |
   |   (same)      |                |               |  |
   +---------------+                +---------------+  |
         |                                    |        |
         | sends (After transfer)             | sends  |
   +----------------+                +----------------+|
   | Media Stream A2|                | Media Stream C1||
   +----------------+                +----------------+|
   | Media StreamRef|                | Media StreamRef||
   | Codec params   |                | Codecparams    ||
   |                |                |                ||
   +----------------+                +----------------+|
           |                                  |        |
           |                                  |        |
           |                                  |        |
           +-------------------------------------------+
```

5.4. Conference Use Cases

Depending on who act as SRC and the information that an SRC has there can be several ways to model conference use cases. This section has instance diagrams for the following cases:

- o A CS where one of the participant (which is also SRC) is a user in a conference
- o A CS where one of the participant is focus (which is also SRC)

- o A CS where one of the participant is user and the SRC is a different entity like B2BUA
- o A CS where one of the participant is focus and the SRC is a different entity like B2BUA

NOTE: There MAY be other ways to model the same use cases depending on what information the SRC has.

5.4.1. Case 1:

This is the usecase where there is a CS with one of the participant (who is also SRC) as a user in a conference. For the sake of simplicity the receive lines for each of the participant is not shown.

```
+-----------------------------------------------------+
|      Communication Session                          |
| +--------------+             +---------------+      |
| |              |             |               |      |
| |Participant B|              | Participant A|       |
| |  (User in    |-------------|               |      |
| |  conf/SRC)   |             |               |      |
| +--------------+             +---------------+      |
|    |  |  |  |                                       |
+-----------------------------------------------------+
     |  |  |  |
     |  |  |  |
     D   E  F  G (Participants of Conference)
```

Instance Diagram:

```
+--------------------------------+
| Recording Session (RS)         |
+--------------------------------+
           |
           |
           |
+--------------------------------+
|   Communication Session (CS)   |
|      Group(CSG)                |
+--------------------------------+
|   Unique-id1                   |
+--------------------------------+
           |
           |
           |
```

```
  +----------------+
  | Communication  |
  | Session (CS)   |--+
  +----------------+  |
  |                |  |
  +----------------+  |
           |          |
           |          |
           |          |
+---------------+     |
| ParticipantA  |     |
|               |     |
+---------------+     |
           |          |
  sends    |          |
           |          |
+---------------+     |
|Media Stream A1|     |
+---------------+     |
|MediaStream Ref|     |
|codec  params  |     |
+---------------+     |
                      |
                      |
        +-------------+
        |          |
        |          |
+----------------+ |
| Participant B  | |
|  (in conf)     | |
+----------------+ |
        |          |
 sends  |  +-------+
        |  |
+----------------+
| Media Stream B1|
+----------------+        sends ( not sure if sends is right word)
| MediaStream Ref|-----+-----------+-------------+---------+
| Codec Params   |     |           |             |         |
+----------------+     |           |             |         |
           +-------------+  +-------------+ +-------------+ +--------------+
           |participantD|  |ParticipantE| |ParticipantF| |Participant G|
           +-------------+  +-------------+ +-------------+ +--------------+
```

In this example we have two participants A and B who are part of a Communication Session(CS). One of the participants B is part of a conference and also acts as SRC.There can be two cases here. B can be a participant of the conference or B can be a focus. In this instance diagram Participant B is a user in a conference. The SRC (Participant B) SHALL subscribe to conference event package to get the details of other particiants. Participant B(SRC) SHALL send the same through the metadata to SRS. In this instance diagram the Media Stream(mixed stream) sent from Participant B SHALL have media streams contributed by conference participants (D,E,F and G). For the sake of simplicity the "receives" line is not shown here. In this example the media stream sent by each participant(A or B) of CS is received by all other participant(A or B).

5.4.2. Case 2:

This is the usecase where there is a CS where one of the participant is focus (which is also SRC).

```
+----------------------------------------------------+
|      Communication Session                         |
| +--------------+              +--------------+     |
| |              |--------------|              |     |
| |Participant C |              | Participant A|     |
| |  (Focus in   |------+       |              |     |
| | conf and SRC)|---+  |       +--------------+     |
| +--------------+   |  |                            |
|   |                |  +---------+                  |
|   |                |            |                  |
| +--------------+   |         +---------------+     |
| | Participant B|   +---+     | Participant D |     |
| |              |       |     |               |     |
| +--------------+       |     +---------------+     |
|                        |                           |
|                   +--------------+                 |
|                   |Participant E |                 |
|                   |              |                 |
|                   +--------------+                 |
|                                                    |
+----------------------------------------------------+
```

Instance Diagram:

```
+-------------------------------+
| Recording Session (RS)        |
+-------------------------------+
           |
           |
```

```
              |
   +-------------------------------+
   |   Communication Session (CS)  |
   |      Group(CSG)               |
   +-------------------------------+
   |   Unique-id1                  |
   +-------------------------------+
              |
              |
              |
     +----------------+
     | Communication  |
     | Session (CS)   |----------------------+
     +----------------+                      |
     |                |                      |
     +----------------+                      |
              |                              |
              |-------------------+          |
              |        |          |          |
   +---------------+   | +---------------+   |
   | ParticipantA  |   | | ParticipantB  |   |
   |               |   | |               |   |
   +---------------+   | +---------------+   |
              |        |         |           |
     sends    |        |         |  sends    |
              |        |         |           |
   +---------------+   |   +---------------+ |
   |Media Stream A1|   |   |Media Stream B1| |
   +---------------+   |   +---------------+ |
   |MediaStream Ref|   |   |MediaStream Ref| |
   |codec  params  |---+---|codec params   | |
   +---------------+       +---------------+ |
                                             |
          +----------------------------------+
          |            |         |           |
          |            |         |           |
   +---------------+   | +---------------+   |
   | ParticipantD  |   | | ParticipantE  |   |
   |               |   | |               |   |
   +---------------+   | +---------------+   |
              |        |         |           |
     sends    |        |         |  sends    |
              |        |         |           |
   +---------------+   |   +---------------+ |
   |Media Stream D1|   |   |Media Stream E1| |
   +---------------+   |   +---------------+ |
   |MediaStream Ref|   |   |MediaStream Ref| |
   |codec  params  |---+---|codec params   | |
```

```
   +----------------+       +----------------+ |
                                               |
                                               |
                                  +-----------+
                +-----------------|
                |                 |
                |                 |
               +-----------------+ |
               | Participant C  | |
               |  (focus +src)  | |
               +-----------------+ |
                       |           |
                Sends  |  +-------+
                       |  |
   "sends" OR          |  |
   contributed +-----------------+
    by         | Media Stream C1|
   Participants+-----------------+  "receives" by participants A,B,D,E
   A,B,D,E     | MediaStream Ref|-----------------------------------
   ------------| Codec Params   |
               +-----------------+
```

In this example we have two participants A and B who are part of a Communication Session(CS). One of the participants (C) is focus of a conference and also acts as SRC. The SRC (Participant C) being the Focus of the conference SHALL have access to the details of other particiants. SRC (Participant C) SHALL send the same through the metadata to SRS. In this instance diagram the Media Stream(mixed stream) sent by C SHALL have media streams contributed by conference participants (A, B, D and E). Participants A, B,D and E SHALL send Media Streams A1, B1, D1 and E1 respectively. The media stream sent by Participant C(Focus) shall be received by all other participants of CS. For the sake of simplicity the "receives" line is not shown linked to all other participants.

NOTE: SRC (Participant C) MAY send mixed stream or seperate streams to SRS

5.4.3. Case 3:

A CS where one of the participant is user and the SRC is a different entity like B2BUA. In this case the SRC MAY not know that one of the user is part of conference. Hence the instance diagram will not have information about the conference participants.

```
   +-----------------------------------------------------+
   |      Communication Session                          |
   | +-------------+  +-------+    +---------------+      |
   | |             |  | (SRC)|    |               |      |
   | |Participant B|--|B2BUA |----| Participant A|      |
   | |  (User in   |  +-------+    |               |      |
   | |  conf)      |               |               |      |
   | +-------------+               +---------------+      |
   |   |  |  |  |                                         |
   +-----------------------------------------------------+
       |  |  |  |
       |  |  |  |
       D   E  F  G (Participants of Conference)
```

5.4.4. Case 4:

A CS where one of the participant is focus and the SRC is a different entity like B2BUA. In this case the participant which is focus MAY send "isfocus" in SIP message to SRC. The SRC MAY subscribe to conference event package on seeing this "isfocus". SRC SHALL learn the details of other participants of conference from the conference package and send the same in metadata to SRS. The instance diagram for this use case SHALL be same as Case 1.

```
              +---------------------------------+
              |   Conference Event Package      |
              |                                 |
              +---------------------------------+
                         |
                         | subscribes
                         |
   +---------------------|-----------------------------+
   |      Communication  |Session                      |
   | +-------------+  +-------+    +---------------+      |
   | |             |  | (SRC)|    |               |      |
   | |Participant B|--|B2BUA |----| Participant A|      |
   | |  (FOCUS in  |  +-------+    |               |      |
   | |  conf)      |               |               |      |
   | +-------------+               +---------------+      |
   |   |  |  |  |                                         |
   +-----------------------------------------------------+
       |  |  |  |
       |  |  |  |
       D   E  F  G (Participants of Conference)
```

6. Security Considerations

The metadata information sent from SRC to SRS MAY reveal sensitive Information about different participants of CS. For this reason, it is RECOMMENDED that a SRC use a strong means for authentication and metadata information protection and that it apply comprehensive authorization rules when using the metadata model defined in this document. The security considerations for this SHALL be defined in the solution document.

7. IANA Considerations

Not Applicable

8. Acknowledgement

We wish to thank John Elwell, Henry Lum, Leon Portman, De Villers, Andrew Hutton, Deepanshu Gautam, Charles Eckel for their valuable comments.

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

9.2. Informative References

[I-D.ietf-siprec-req]
Rehor, K., Portman, L., Hutton, A., and R. Jain, "Requirements for SIP-based Media Recording (SIPREC)", draft-ietf-siprec-req-06 (work in progress), December 2010.

[I-D.ietf-siprec-architecture]
Hutton, A., Portman, L., Jain, R., and K. Rehor, "An Architecture for Media Recording using the Session Initiation Protocol", draft-ietf-siprec-architecture-01 (work in progress), October 2010.

Authors' Addresses

Ram Mohan R
Cisco Systems, Inc.
Cessna Business Park,
Kadabeesanahalli Village, Varthur Hobli,
Sarjapur-Marathahalli Outer Ring Road
Bangalore, Karnataka 560103
India

Email: rmohanr@cisco.com

Parthasarathi R
Cisco Systems, Inc.
Cessna Business Park,
Kadabeesanahalli Village, Varthur Hobli,
Sarjapur-Marathahalli Outer Ring Road
Bangalore, Karnataka 560103
India

Email: partr@cisco.com

P. Kyzivat
Cisco Systems, Inc.
1414 Massachusetts Avenue
Boxborough, MA 01719
USA

Email: pkyzivat@cisco.com

siprec   Ram Mohan. Ravindranath
Internet-Draft   Parthasarathi. Ravindran
Intended status: Standards Track   Paul. Kyzivat
Expires: December 19, 2011   Cisco Systems, Inc.
                                                                June 17, 2011

Session Initiation Protocol (SIP) Recording Metadata Format
draft-ram-siprec-metadata-format-02

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 focuses on the Recording metadata format which describes the communication session.

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. The requirements for such recording is described in [I-D.ietf-siprec-req], the related architecture is described in [I-D.ietf-siprec-architecture], and the metadata model viewed by Session Recording Server is described in [I-D.ietf-siprec-metadata]. This document focuses on the Recording metadata format which describes the communication session. The delivery mechanism for passing metadata is outside the scope of this document.

The Session Recording Client (SRC) initiates the Recording Session. Here, Recording Session is a completely independent from the Communication Session that is being recorded at both the SIP dialog level and at the session level.

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 [RFC2119]. This document only uses these key words when referencing normative statements in existing RFCs.

3. Recording Metadata Format

Recording Metadata is the data that describes the communication session. Metadata has to be conveyed from SRC to SRS, further the metadata MAY be conveyed in the Recording Session dialog. Few metadata information SHALL be derived from RS dialog. Recording dialog-id SHALL be used as recording specific unique id, Date header SHALL be used as start and stop time of recording metadata block.

The media related details of metadata SHALL be passed across using session description protocol (SDP) [RFC4566]. SDP attributes describes about different media formats like audio, video. The other metadata attributes like participant details MUST be passed across in new Recording specific XML document namely application/rs-metadata+xml. The linkage between application/rs-metadata+xml XML schema and metadata SDP is done using the SDP label attribute (a=label:xxx) referenced in [RFC4574].

Metadata is passed across in Recording Session(RS) incrementally whenever there is a change in CS.

4. SIP Recording Metadata document format

4.1. Contents

Recording Metadata document is an XML document which will be embedded as a message body. The document contains

- o recording element MUST present in all recording metadata XML document. recording acts as container for all other elements in this XML document.
- o Elements like session, participant, stream and group are under recording element directly with appropriate parent id and have separate URN UUID for passing the partial information of metadata. In case of partial metadata, recording element and the relevant updated elements will be passed by SRC and the elements are identified in SRS using URN UUID and parent id.
- o Group element is an optional element provides the information about the communication session group
- o Session element provides the information about the communication session
- o Participant element provides information regarding the specific participant involved in the recording
- o Stream element indicates SDP media lines associated with the session and participants
- o Extensiondata element provides the mechanism by which namespace/ element MAY be extended with standard or proprietary information.

4.2. XML data format

Recording object is a XML document. It MUST have the XML declaration and it SHOULD contain an encoding declaration in the XML declaration, e.g., "<?xml version='1.0' encoding='UTF-8'?>". If the charset parameter of the MIME content type declaration is present and it is different from the encoding declaration, the charset parameter takes precedence.

Every application conforms to this specification MUST accept the UTF-8 character encoding to ensure the minimal interoperability.

Syntax and semantics error in recording XML document has to be informed to the originator using application specific mechanism.

4.2.1. Namespace

The namespace URI for elements defined by this specification is a Uniform Resource Namespace (URN) [RFC2141], using the namespace identifier 'ietf' defined by [RFC2648] and extended by [RFC3688].

The URN is as follows: urn:ietf:params:xml:ns:recording

4.2.2. recording

recording element MUST contain an xmlns namespace attribute with value as urn:ietf:params:xml:ns:siprec. One recording element MUST present in the all recording metadata XML document.

recording element has group, session, stream, participant elements.

dataMode element shows whether the XML document is complete document or partial update. The default value is complete.

4.2.3. group

Each communication session group (CSG) is represented using one group element. Each group element has unique URN UUID attribute which helps to uniquely identify CSG.

4.2.4. session

Each communication session(CS) has one session element. Each session element has unique URN UUID attribute which helps to uniquely identify CS.

Reason element MAY be included to indicate the reason for termination. group-ref element MAY exist to indicate the group where the mentioned session belongs.

4.2.5. participant

Each communication session user is defined by one participant element and there MUST be atleast 2 participant for any given session. "send" or "receive" element in each participant is associating SDP m-lines with the participant. send element indicates that participant is sending the stream of media with the mentioned media description. recv element indicates that participant is receiving the stream and by default all participant will receive the stream. recv element has relevance in case whisper call scenario wherein few of the participant in the session receives the stream and not others.

Participant MUST have AOR element which contains SIP/SIPS URI to identify the participant. AOR element is SIP/SIPS URI FQDN or IP address which represents the user. name is an optional element to represent display name.

Each participant element has unique URN UUID attribute which helps to uniquely identify participant and session URN UUID to associate participant with specific session element. URN UUID of participant *MUST* used in the scope of CSG and no new URN UUID has to be created for the same element (participant, stream) between different CS in the same CSG. In case URN UUID has to be used permanent, careful usage of URN UUID to original AoR has to be decided by the implementers and it is implementer’s choice.

4.2.6. stream

This element indicates the SDP m-line properties like label attributes, media mode. Label attribute is used to link m-line SDP body using label attribute in SDP m-line. The media mode helps in understanding whether the media is mixed or not.

Each stream element has unique URN UUID attribute which helps to uniquely identify stream and session URN UUID to associate stream with specific session element. The open item here is whether to use URN UUID (global id) or xml:id (local id).

4.2.7. extensiondata

extensiondata element SHALL include any other XML namespace. Multiple namespace MAY exists under extensiondata. extensiondata element exist in each level like recording, session, participant, stream to provide extensiondata specific to each element. extensiondata element has unique id based on URN UUID [RFC4122] attribute and its parent id. The open item in extensiondata is whether any need of separate metadata block or not.

4.2.8. start-time/stop-time

start-time/stop-time contains a string indicating the date and time of the status change of this tuple. The value of this element MUST follow the IMPP datetime format [RFC3339]. Timestamps that contain ’T’ or ’Z’ MUST use the capitalized forms. At a time, any of the time tuple start-time or stop-time MAY exist in the element namely group, session, participant, stream and not both timestamp at the same time.

As a security measure, the timestamp element SHOULD be included in all tuples unless the exact time of the status change cannot be determined.

5. SIP Recording Metadata Example

5.1. Complete SIP Recording Metadata Example

The following example provides all the tuples involved in Recording Metadata XML body.


```
<?xml version="1.0" encoding="UTF-8"?>
<recording xmlns='urn:ietf:params:xml:ns:recording'>
   <group id="urn:uuid:efe3930b-2a31-4e6a-a6ab-203fd7078302">
     <start-time>2010-12-16T23:41:07Z</start-time>
   </group>
   <extensiondata id="urn:uuid:f3373a7b-4958-4e55-8820-d03a191fb76a"
     parent="urn:uuid:efe3930b-2a31-4e6a-a6ab-203fd7078302">
     <!-- Standardized extension -->
     <call-center xmlns='urn:ietf:params:xml:ns:callcenter'>
       <supervisor>sip:alice@cisco.com</supervisor>
     </call-center>
     <mydata xmlns='http://example.com/my'>
       <structure>FOO!</structure>
       <whatever>bar</whatever>
     </mydata>
   </extensiondata>
   <session id="urn:uuid:855a5ded-8420-456d-a70f-6da1eeaeb425">
     <group-ref>urn:uuid:efe3930b-2a31-4e6a-a6ab-203fd7078302
     </group-ref>
     <start-time>2010-12-16T23:41:07Z</start-time>
   </session>
   <extensiondata id="urn:uuid:a54d6aa5-d40d-43f9-88c5-b4633d873bdd"
     parent="urn:uuid:855a5ded-8420-456d-a70f-6da1eeaeb425">
     <structure>FOO!</structure>
     <whatever>bar</whatever>
   </extensiondata>
   <participant
      id="urn:uuid:b2b7c112-5982-469d-9007-6ddbecca64d3"
      session="urn:uuid:855a5ded-8420-456d-a70f-6da1eeaeb425">
      <aor>sip:partha@blr.cisco.com</aor>
      <send>urn:uuid:8b53f3de-da39-4846-93c7-ee5e5f8f6f0b</send>
      <send>urn:uuid:50000c9b-9191-40a4-8231-5bcbca5e2b17</send>
      <start-time>2010-12-16T23:41:07Z</start-time>
   </participant>
   <extensiondata
     id="urn:uuid:7edca82f-054d-47f2-a032-9b2a5b5186c1"
     parent="urn:uuid:b2b7c112-5982-469d-9007-6ddbecca64d3">
     <structure>FOO!</structure>
     <whatever>bar</whatever>
   </extensiondata>
   <participant
```

```
           id="urn:uuid:cd27cfa1-2bdd-4830-a653-70374d10f103"
           session="urn:uuid:855a5ded-8420-456d-a70f-6da1eeaeb425">
           <aor>sip:paul@box.cisco.com</aor>
           <send>urn:uuid:50000c9b-9191-40a4-8231-5bcbca5e2b17</send>
           <send>urn:uuid:8b53f3de-da39-4846-93c7-ee5e5f8f6f0b</send>
           <start-time>2010-12-16T23:41:07Z</start-time>
       </participant>
       <extensiondata
         id="urn:uuid:60a76c80-d399-11d9-b91C-0003939e0af6"
         parent="urn:uuid:cd27cfa1-2bdd-4830-a653-70374d10f103">
         <structure>FOO!</structure>
         <whatever>bar</whatever>
       </extensiondata>
       <stream id="urn:uuid:50000c9b-9191-40a4-8231-5bcbca5e2b17"
           session="urn:uuid:855a5ded-8420-456d-a70f-6da1eeaeb425">
           <start-time>2010-12-16T23:41:07Z</start-time>
           <label>96</label>
       </stream>
       <stream id="urn:uuid:8b53f3de-da39-4846-93c7-ee5e5f8f6f0b"
           session="urn:uuid:855a5ded-8420-456d-a70f-6da1eeaeb425">
           <start-time>2010-12-16T23:41:07Z</start-time>
           <label>97</label>
       </stream>
       <stream id="urn:uuid:f3373a7b-4958-4e55-8820-d03a191fb76a"
           session="urn:uuid:cd27cfa1-2bdd-4830-a653-70374d10f103">
           <start-time>2010-12-16T23:41:07Z</start-time>
           <label>98</label>
       </stream>
       <stream id="urn:uuid:1225c695-cfb8-4ebb-aaaa-80da344efa6a"
           session="urn:uuid:cd27cfa1-2bdd-4830-a653-70374d10f103">
           <start-time>2010-12-16T23:41:07Z</start-time>
           <label>99</label>
       </stream>
  </recording>
```


SIP Recording Metadata Example XML body

5.2. Partial Update of Recording metadata XML body

The following example provides partial update in Recording Metadata XML body for the above example. The example illustrate the stop time of the specific stream.

```
     <?xml version="1.0" encoding="UTF-8"?>
     <recording xmlns='urn:ietf:params:xml:ns:siprec'>
       <dataMode>partial</dataMode>
       <stream id="urn:uuid:50000c9b-9191-40a4-8231-5bcbca5e2b17"
            session="urn:uuid:855a5ded-8420-456d-a70f-6da1eeaeb425">
            <label>96</label>
        </stream>
        <stream id="urn:uuid:8b53f3de-da39-4846-93c7-ee5e5f8f6f0b"
            session="urn:uuid:855a5ded-8420-456d-a70f-6da1eeaeb425">
            <label>97</label>
        </stream>
        <stream id="urn:uuid:f3373a7b-4958-4e55-8820-d03a191fb76a"
            session="urn:uuid:cd27cfa1-2bdd-4830-a653-70374d10f103">
            <label>98</label>
        </stream>
        <stream id="urn:uuid:1225c695-cfb8-4ebb-aaaa-80da344efa6a"
            session="urn:uuid:cd27cfa1-2bdd-4830-a653-70374d10f103">
            <label>99</label>
        </stream>
     </recording>
```

Partial update of SIP Recording Example XML body

6. XML Schema definition for Recording metadata

This section defines XML schema for Recording metadata document

```
   <?xml version="1.0" encoding="UTF-8"?>
   <xs:schema targetNamespace="urn:ietf:params:xml:ns:recording"
    xmlns:tns="urn:ietf:params:xml:ns:recording"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    elementFormDefault="qualified"
    attributeFormDefault="unqualified">
   <!-- This import brings in the XML language attribute xml:lang-->
   <xs:import namespace="http://www.w3.org/XML/1998/namespace"
    schemaLocation="http://www.w3.org/2001/xml.xsd"/>
   <xs:element name="recording" type="recording"/>
   <xs:complexType name="recording">
       <xs:sequence>
           <xs:element name="datamode" type="dataMode"
                minOccurs="0"/>
           <xs:element name="session" type="session"
                minOccurs="0" maxOccurs="unbounded"/>
           <xs:element name="participant" type="participant"
                minOccurs="0" maxOccurs="unbounded"/>
           <xs:element name="stream" type="stream"
               minOccurs="0" maxOccurs="unbounded"/>
```

```
         <xs:element name="extensiondata" type="extensiondata"
               minOccurs="0" maxOccurs="unbounded"/>
       </xs:sequence>
   </xs:complexType>
   <xs:complexType name="group">
       <xs:sequence>
         <xs:element name="start-time" type="xs:dateTime"
            minOccurs="0"/>
         <xs:element name="stop-time" type="xs:dateTime"
            minOccurs="0"/>
       </xs:sequence>
       <xs:attribute name="id" type="urnuuid"
            use="required"/>
   </xs:complexType>
   <xs:complexType name="session">
       <xs:sequence>
         <xs:element name="start-time" type="dateTime"
                minOccurs="0"/>
         <xs:element name="stop-time" type="dateTime"
                minOccurs="0"/>
         <xs:element name="reason" type="xs:string"
                minOccurs="0"/>
         <xs:element name="group-ref" type="urnuuid"
                minOccurs="0" maxOccurs="1"/>
       </xs:sequence>
       <xs:attribute name="id" type="urnuuid"
            use="required"/>
   </xs:complexType>
   <xs:complexType name="participant">
       <xs:sequence>
           <xs:element name="aor" type="xs:anyURI"
             maxOccurs="1"/>
         <xs:element name="name" type="xs:string"
             minOccurs="0" maxOccurs="1"/>
           <xs:element name="send" type="urnuuid"
             minOccurs="0" maxOccurs="unbounded"/>
           <xs:element name="recv" type="urnuuid"
             minOccurs="0" maxOccurs="unbounded"/>
         <xs:element name="start-time" type="xs:dateTime"
                minOccurs="0"/>
         <xs:element name="stop-time" type="xs:dateTime"
                minOccurs="0"/>
       </xs:sequence>
       <xs:attribute name="id" type="urnuuid"
            use="required"/>
       <xs:attribute name="session" type="urnuuid"
            use="required"/>
   </xs:complexType>
```

```
   <xs:complexType name="stream">
       <xs:sequence>
         <xs:element name="label" type="xs:string"
             minOccurs="0" maxOccurs="1"/>
           <xs:element name="mode" type="streamMode"
             minOccurs="0" maxOccurs="1"/>
         <xs:element name="start-time" type="xs:dateTime"
                minOccurs="0"/>
         <xs:element name="stop-time" type="xs:dateTime"
                minOccurs="0"/>
       <xs:element name="extensiondata" type="extensiondata"
             minOccurs="0"/>
       </xs:sequence>
       <xs:attribute name="id" type="urnuuid"
            use="required"/>
       <xs:attribute name="session" type="urnuuid"
            use="required"/>
   </xs:complexType>
   <xs:element name='extensiondata'>
     <xs:complexType>
         <xs:any     namespace='##other'
                     minOccurs='0'
                     maxOccurs='unbounded'
                     processContents='lax'/>
     </xs:complexType>
     <xs:attribute name="id" type="urnuuid"
            use="required"/>
     <xs:attribute name="parent" type="urnuuid"
       use="required"/>
   </xs:element>
   <xs:simpleType name="streamMode">
       <xs:restriction base="xs:string">
           <xs:pattern
            value="mixed|separate"/>
       </xs:restriction>
   <xs:simpleType name="urnuuid">
         <xs:restriction base="xs:string">
           <xs:pattern
            value="urn:uuid:[0-9a-zA-Z]{8}-[0-9a-zA-Z]{4}
                 -[0-9a-zA-Z]{4}-[0-9a-zA-Z]{4}-[0-9a-zA-Z]{12}"/>
         </xs:restriction>
   </xs:simpleType>
   </xs:simpleType>
     <xs:simpleType name="dataMode">
         <xs:restriction base="xs:string">
           <xs:pattern
            value="complete|partial"/>
         </xs:restriction>
```

```
</xs:simpleType>
```

7. Example with SIP and metadata XML+SDP

This section describes the different use cases/messages for delivering Metadata in a Recording Sessions. This section is written in the draft for better readability and the example will be moved to solution document or removed when this draft is adopted as WG item.

7.1. SRC Initiated Recording

An SRC initiates Recording Session(RS) for recording a communication session with audio and video media. SRC initiates the dialog by sending an INVITE request to the SRS. INVITE is formed as specified in [RFC3261] , SRC inserts recording metadata as an XML document and SDP in multipart MIME message body [RFC2046]. The content type of SIP header is set to application/rs-metadata+xml [I-D.portman-siprec-protocol]. SRC MUST form SDP offer using the normal procedures defined in [RFC3261]and [RFC3264]. SRC SHALL include one m-line for each stream of each participant. If the recording has to be started immediately then SRC MUST include an SDP attribute of "a=sendonly" for each media line or "a=inactive" if it is not ready to transmit the media. SRC MAY also include only one m-line for all streams of same type for all participants depending on whether it has the capability to mix the streams. SRC indicates the modes (mixed or single) for each stream using a mode attribute. An example wherein INVITE sent by an SRC is shown below:

```
INVITE sip:1041@recordingserver.cisco.com:5060;transport=tcp SIP/2.0
Via: SIP/2.0/TCP 192.0.2.58;branch=z9hG4bK-19935-1-7
Max-Forwards: 70
To: <sip:1041@recordingserver.cisco.com>
From: RecrdingClient <sip:192.0.2.58>;tag=ds43d76263
Call-ID: 12548086970261@192.0.2.58
CSeq: 100 INVITE
Content-Length: xxx
Contact: <sip:192.0.2.58:5060;transport=tcp>
Date: Tue, 16 Dec 2010 23:41:07 GMT
Content-Type: multipart/mixed;boundary=unique-boundary-1
MIME-Version: 1.0

--unique-boundary-1
Content-Type: application/SDP
...
m=audio 49170 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

```
   a=label:96
   a=sendonly
   ...
   m=video 49174 RTP/AVPF 96
   a=rtpmap:96 H.264/90000
   a=label:97
   a=sendonly
   ...
   m=audio 51372 RTP/AVP 0
   a=rtpmap:0 PCMU/8000
   a=label:98
   a=sendonly
   ...
   m=video 49176 RTP/AVPF 96
   a=rtpmap:96 H.264/90000
   a=label:99
   a=sendonly
   ....

   --unique-boundary-1
   Content-type:application/rs-metadata+xml

   <?xml version="1.0" encoding="UTF-8"?>
     <recording xmlns='urn:ietf:params:xml:ns:recording'>
        <group id="urn:uuid:efe3930b-2a31-4e6a-a6ab-203fd7078302">
        </group>
        <session id="urn:uuid:855a5ded-8420-456d-a70f-6da1eeaeb425">
          <group-ref>urn:uuid:efe3930b-2a31-4e6a-a6ab-203fd7078302
          </group-ref>
          <start-time>2010-12-16T23:41:07Z</start-time>
        </session>
        <participant
           id="urn:uuid:b2b7c112-5982-469d-9007-6ddbecca64d3"
           session="urn:uuid:855a5ded-8420-456d-a70f-6da1eeaeb425">
           <aor>sip:partha@blr.cisco.com</aor>
           <send>urn:uuid:8b53f3de-da39-4846-93c7-ee5e5f8f6f0b</send>
           <send>urn:uuid:50000c9b-9191-40a4-8231-5bcbca5e2b17</send>
           <start-time>2010-12-16T23:41:07Z</start-time>
        </participant>
        <participant
            id="urn:uuid:cd27cfa1-2bdd-4830-a653-70374d10f103"
            session="urn:uuid:855a5ded-8420-456d-a70f-6da1eeaeb425">
            <aor>sip:paul@box.cisco.com</aor>
            <send>urn:uuid:50000c9b-9191-40a4-8231-5bcbca5e2b17</send>
            <send>urn:uuid:8b53f3de-da39-4846-93c7-ee5e5f8f6f0b</send>
            <start-time>2010-12-16T23:41:07Z</start-time>
        </participant>
        <stream id="urn:uuid:50000c9b-9191-40a4-8231-5bcbca5e2b17"
```

```
            session="urn:uuid:855a5ded-8420-456d-a70f-6da1eeaeb425">
            <start-time>2010-12-16T23:41:07Z</start-time>
            <label>96</label>
        </stream>
        <stream id="urn:uuid:8b53f3de-da39-4846-93c7-ee5e5f8f6f0b"
            session="urn:uuid:855a5ded-8420-456d-a70f-6da1eeaeb425">
            <start-time>2010-12-16T23:41:07Z</start-time>
            <label>97</label>
        </stream>
        <stream id="urn:uuid:f3373a7b-4958-4e55-8820-d03a191fb76a"
            session="urn:uuid:cd27cfa1-2bdd-4830-a653-70374d10f103">
            <start-time>2010-12-16T23:41:07Z</start-time>
            <label>98</label>
        </stream>
        <stream id="urn:uuid:1225c695-cfb8-4ebb-aaaa-80da344efa6a"
            session="urn:uuid:cd27cfa1-2bdd-4830-a653-70374d10f103">
            <start-time>2010-12-16T23:41:07Z</start-time>
            <label>99</label>
        </stream>
   </recording>

   --unique-boundary-1--
```


7.2. SRC updates about participant change

An SRC updates about participant change without impact any change in MS, CSG using RE-INVITE. An example wherein RE-INVITE sent by an SRC for the participant update is shown below:

```
INVITE sip:1041@recordingserver.cisco.com:5060;transport=tcp SIP/2.0
Via: SIP/2.0/TCP 192.0.2.58;branch=z9hG4bK-19935-1-7
Max-Forwards: 70
To: <sip:1041@recordingserver.cisco.com>
From: RecrdingClient <sip:192.0.2.58>;tag=ds43d76263
Call-ID: 12548086970261@192.0.2.58
CSeq: 100 INVITE
Content-Length: xxx
Contact: <sip:192.0.2.58:5060;transport=tcp>
Date: Tue, 16 Dec 2010 23:41:07 GMT
Content-Type: multipart/mixed;boundary=unique-boundary-1
MIME-Version: 1.0

--unique-boundary-1
Content-Type: application/SDP
...
```

```
m=audio 49170 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=label:96
a=sendonly
...
m=video 49174 RTP/AVPF 96
a=rtpmap:96 H.264/90000
a=label:97
a=sendonly
...
m=audio 51372 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=label:98
a=sendonly
...
m=video 49176 RTP/AVPF 96
a=rtpmap:96 H.264/90000
a=label:99
a=sendonly
....

--unique-boundary-1
Content-type:application/rs-metadata+xml
```


```
<?xml version="1.0" encoding="UTF-8"?>
  <recording xmlns='urn:ietf:params:xml:ns:recording'>
     <dataMode>partial</dataMode>
     <session id="urn:uuid:18bba4ff2-9663-11e0-9516-5b9b4824019b">
       <group-ref>urn:uuid:efe3930b-2a31-4e6a-a6ab-203fd7078302
       </group-ref>
       <start-time>2010-12-16T23:45:07Z</start-time>
     </session>
     <session id="urn:uuid:855a5ded-8420-456d-a70f-6da1eeaeb425">
       <stop-time>2010-12-16T23:45:07Z</stop-time>
     </session>
     <participant
        id="urn:uuid:b2b7c112-5982-469d-9007-6ddbecca64d3"
        session="urn:uuid:18bba4ff2-9663-11e0-9516-5b9b4824019b">
        <aor>sip:partha@blr.cisco.com</aor>
        <send>urn:uuid:8b53f3de-da39-4846-93c7-ee5e5f8f6f0b</send>
        <send>urn:uuid:50000c9b-9191-40a4-8231-5bcbca5e2b17</send>
     </participant>
     <participant
         id="urn:uuid:cd27cfa1-2bdd-4830-a653-70374d10f103"
         session="urn:uuid:855a5ded-8420-456d-a70f-6da1eeaeb425">
         <stop-time>2010-12-16T23:45:07Z</stop-time>
     </participant>
     <participant
```

```
            id="urn:uuid:cd27cfa1-2bdd-4830-a653-70374d10f103"
            session="urn:uuid:18bba4ff2-9663-11e0-9516-5b9b4824019b">
            <start-time>2010-12-16T23:45:07Z</stop-time>
            <aor>sip:ram@blr.cisco.com</aor>
            <send>urn:uuid:50000c9b-9191-40a4-8231-5bcbca5e2b17</send>
            <send>urn:uuid:8b53f3de-da39-4846-93c7-ee5e5f8f6f0b</send>
        </participant>
        <stream id="urn:uuid:50000c9b-9191-40a4-8231-5bcbca5e2b17"
            session="urn:uuid:18bba4ff2-9663-11e0-9516-5b9b4824019b">
            <label>96</label>
        </stream>
        <stream id="urn:uuid:8b53f3de-da39-4846-93c7-ee5e5f8f6f0b"
            session="urn:uuid:18bba4ff2-9663-11e0-9516-5b9b4824019b">
            <label>97</label>
        </stream>
        <stream id="urn:uuid:f3373a7b-4958-4e55-8820-d03a191fb76a"
            session="urn:uuid:18bba4ff2-9663-11e0-9516-5b9b4824019b">
            <label>98</label>
        </stream>
        <stream id="urn:uuid:1225c695-cfb8-4ebb-aaaa-80da344efa6a"
            session="urn:uuid:18bba4ff2-9663-11e0-9516-5b9b4824019b">
            <label>99</label>
        </stream>
   </recording>

   --unique-boundary-1--
```


8. Security Considerations

The metadata information sent from SRC to SRS MAY reveal sensitive information about different participants in a session. For this reason, it is RECOMMENDED that a SRC use a strong means for authentication and metadata information protection and that it apply comprehensive authorization rules when using the metadata format defined in this document. The following sections will discuss each of these aspects in more detail.

8.1. Connection Security

It is RECOMMENDED that a SRC authenticate SRS using the normal SIP authentication mechanisms, such as Digest as defined in Section 22 of [RFC3261]. The mechanism used for conveying the metadata information MUST ensure integrity and SHOULD ensure confidentially of the information. In order to achieve these, an end-to-end SIP encryption mechanism, such as S/MIME described in [RFC3261], SHOULD be used.

If a strong end-to-end security means (such as above) is not available, it is RECOMMENDED that a SRC use mutual hop-by-hop Transport Layer Security (TLS) authentication and encryption mechanisms described in "SIPS URI Scheme" and "Interdomain Requests" of [RFC3261].

TBD: Other detailed security aspects

9. IANA Considerations

This specification registers a new XML namespace, and a new XML schema.

9.1. SIP recording metadata Schema Registration

URI: urn:ietf:params:xml:ns:recording

Registrant Contact: IETF SIPREC working group, Ram mohan R(rmohanr@cisco.com)

XML: the XML schema to be registered is contained in Section 6.

Its first line is <?xml version="1.0" encoding="UTF-8"?> and its last line is </xs:schema>

10. Acknowledgement

We wish to thank Joe Hildebrand(Cisco), Peter Saint-Andre(Cisco) for the valuable XML related guidance. Thanks to Michael Benenson(Cisco), Leon Portman(Nice), Henry Lum(Alcatel-lucent), John Elwell(Siemens-Enterprise), Hadriel Kaplan (ACME), Brian Rosen(Neustar), Scott Orton(Broadsoft), Charles Eckel(Cisco) for their inputs and comments.

11. References

11.1. Normative References

[RFC2046] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types", RFC 2046, November 1996.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2141] Moats, R., "URN Syntax", RFC 2141, May 1997.

[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.

[RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, June 2002.

[RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, January 2004.

[RFC3339] Klyne, G., Ed. and C. Newman, "Date and Time on the Internet: Timestamps", RFC 3339, July 2002.

[RFC4122] Leach, P., Mealling, M., and R. Salz, "A Universally Unique IDentifier (UUID) URN Namespace", RFC 4122, July 2005.

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

[RFC4574] Levin, O. and G. Camarillo, "The Session Description Protocol (SDP) Label Attribute", RFC 4574, August 2006.

11.2. Informative References

[I-D.ietf-siprec-req]
Rehor, K., Portman, L., Hutton, A., and R. Jain, "Use Cases and Requirements for SIP-based Media Recording (SIPREC)", draft-ietf-siprec-req-12 (work in progress), June 2011.

[I-D.ietf-siprec-architecture]
Hutton, A., Portman, L., Jain, R., and K. Rehor, "An Architecture for Media Recording using the Session Initiation Protocol", draft-ietf-siprec-architecture-02 (work in progress), April 2011.

[I-D.ietf-siprec-metadata]
R, R., R, P., and P. Kyzivat, "Session Initiation Protocol (SIP) Recording Metadata", draft-ietf-siprec-metadata-00 (work in progress), April 2011.

[I-D.portman-siprec-protocol]
Portman, L., Lum, H., Johnston, A., and A. Hutton,

"Session Recording Protocol",
draft-portman-siprec-protocol-04 (work in progress),
May 2011.

[RFC2648] Moats, R., "A URN Namespace for IETF Documents", RFC 2648,
August 1999.

Authors' Addresses

Ram Mohan Ravindranath
Cisco Systems, Inc.
Cessna Business Park,
Kadabeesanahalli Village, Varthur Hobli,
Sarjapur-Marathahalli Outer Ring Road
Bangalore, Karnataka 560103
India

Email: rmohanr@cisco.com

Parthasarathi Ravindran
Cisco Systems, Inc.
Cessna Business Park,
Kadabeesanahalli Village, Varthur Hobli,
Sarjapur-Marathahalli Outer Ring Road
Bangalore, Karnataka 560103
India

Email: partr@cisco.com

P. Kyzivat
Cisco Systems, Inc.
1414 Massachusetts Avenue
Boxborough, MA 01719
USA

Email: pkyzivat@cisco.com