

SIPREC	L.P. Portman, Ed.
Internet-Draft	NICE Systems
Intended status: Standards Track	H. Lum, Ed.
Expires: February 16, 2012	Genesys, Alcatel-Lucent
	A. Johnston
	Avaya
	A. Hutton
	Siemens Enterprise Communications
	August 15, 2011

Session Recording Protocol
draft-ietf-siprec-protocol-00

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).

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet- Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on February 16, 2012.

Copyright Notice

Copyright (c) 2011 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

- *1. [Introduction](#)
- *2. [Definitions](#)
- *3. [Scope](#)
- *4. [Overview of operations](#)
 - *4.1. [Delivering recorded media](#)
 - *4.2. [Conference focus as an SRC](#)
 - *4.3. [Delivering recording metadata](#)
- *5. [SIP Extensions for Recording Session](#)
 - *5.1. [Callee Capabilities Extensions for SIP Recording](#)
 - *5.1.1. [src Feature Tag](#)
 - *5.1.2. [srs Feature Tag](#)
 - *5.2. [recording-session Options Tag](#)
 - *5.3. [SDP handling](#)
 - *5.4. [RTP handling](#)
 - *5.5. [Metadata](#)
 - *5.6. [Requesting for metadata snapshot](#)
 - *5.6.1. [Formal Syntax](#)
 - *5.7. [Recording Pause and Resume](#)
- *6. [Extensions for Recording-aware User Agents](#)
 - *6.1. [SIP Extensions](#)
 - *6.1.1. [Recording awareness](#)
 - *6.2. [SDP Extensions](#)
 - *6.2.1. [Providing recording indication](#)
 - *6.2.2. [Recording preference](#)
- *7. [IANA Considerations](#)

- *7.1. [Registration of Option Tags](#)
 - *7.1.1. [recording-session Option Tag](#)
 - *7.1.2. [record-aware Option Tag](#)
- *7.2. [Registration of media feature tags](#)
 - *7.2.1. [src feature tag](#)
 - *7.2.2. [srs feature tag](#)
- *7.3. [New Content-Disposition Parameter Registrations](#)
- *7.4. [Media Type Registration](#)
 - *7.4.1. [Registration of MIME Type application/rs-metadata](#)
 - *7.4.2. [Registration of MIME Type application/rs-metadata-request](#)
- *7.5. [SDP Attributes](#)
 - *7.5.1. ['record' SDP Attribute](#)
 - *7.5.2. ['recordpref' SDP Attribute](#)
- *8. [Security Considerations](#)
 - *8.1. [Authentication and Authorization](#)
- *9. [Acknowledgements](#)
- *10. [References](#)
 - *10.1. [Normative References](#)
 - *10.2. [Informative References](#)
- *[Authors' Addresses](#)

[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 [\[RFC6341\]](#) 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 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

This document refers to the core definitions provided in the architecture document [\[I-D.ietf-siprec-architecture\]](#).

3. Scope

The scope of the Session Recording Protocol includes the establishment of the recording sessions and the reporting of the metadata. The scope also includes extensions supported by Record-aware User Agents such as indication of recording. The following items, which are not an exhaustive list, do not represent the protocol itself and are considered out of the scope of the Session Recording Protocol:

- *Recording policies that determine whether the CS should be recorded
- *Retention policies that determine how long a recording is stored
- *Searching and accessing the recorded media and metadata
- *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 SRC is establishing the Recording Session (message #5) after the Communication Session is established.

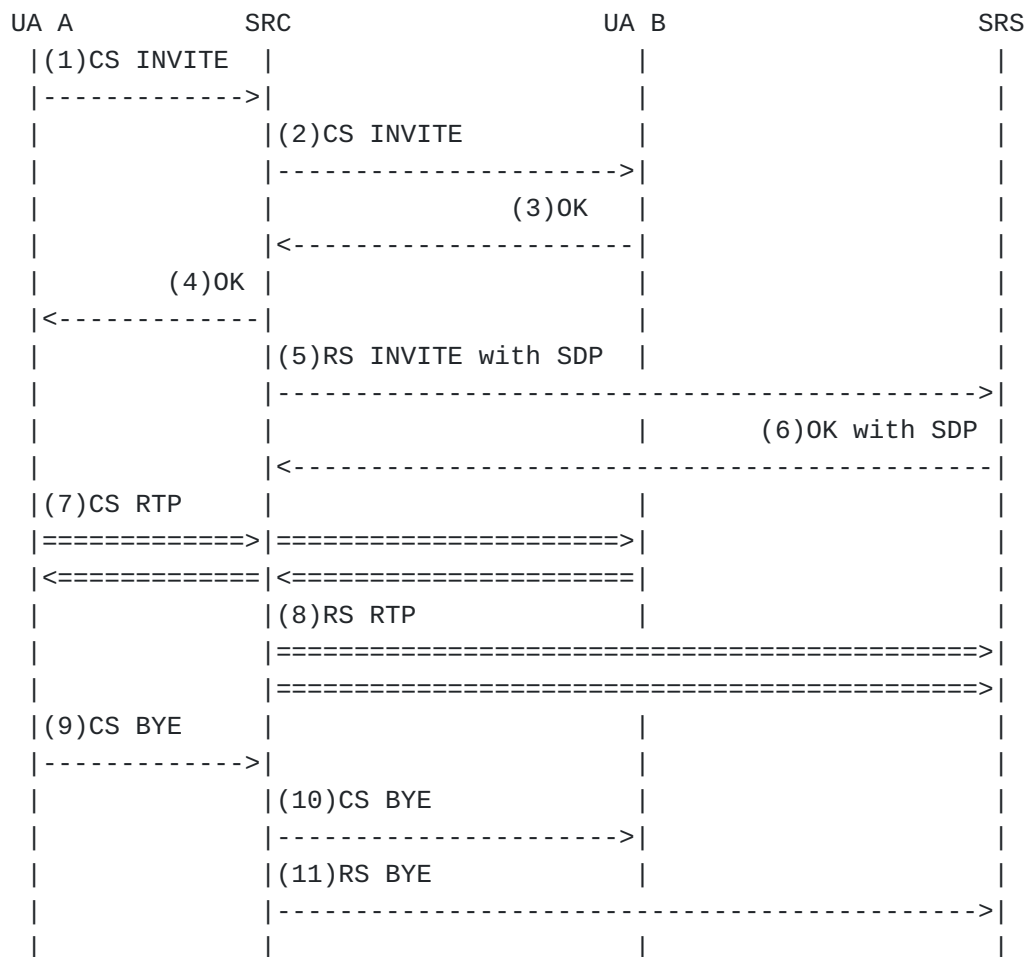


Figure 1: Basic Recording Call flow

[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 [\[RFC4508\]](#).



Figure 2: 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 send 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.

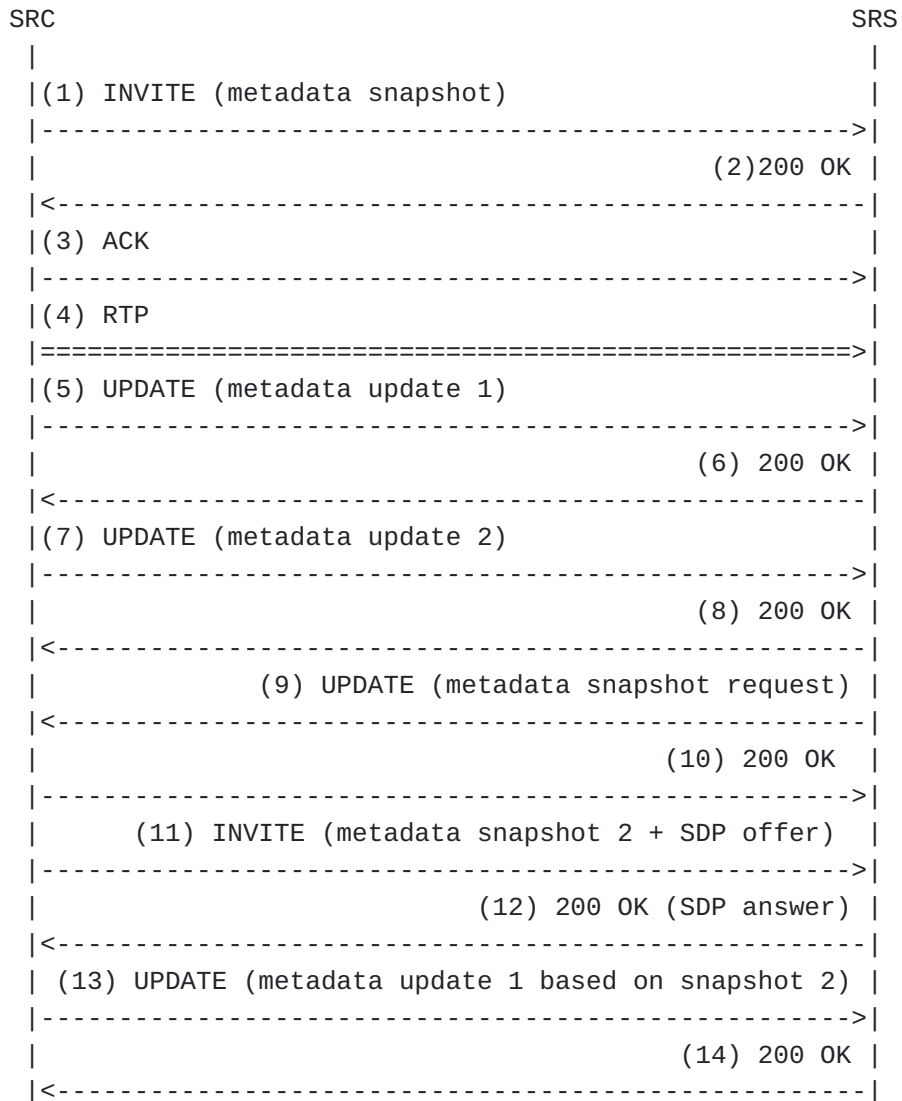


Figure 3: Delivering metadata via SIP UPDATE

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 or the SRS. 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 [\[RFC6341\]](#), 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. Two new feature tags are introduced: '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.

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 in the request.

[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 SRC, an SRS can utilize the SIP Caller Preferences extensions, defined in [\[RFC3841\]](#). The presence of a `Accept-Contact: *;sip.src` allows a UA to request that the INVITE be routed to an SRC. Note that to be completely sure, the SRS would need to include a `Require: prefs` header field in the request.

[5.2.](#) recording-session Options Tag

Since SIP Caller Preferences extensions are optional to implement for routing proxies, there is no guarantee that a recording session will be routed an SRC or SRS. We introduce the use of the recording-session option tag as a mechanism to ensure only an SRC or an SRS would be able to accept recording sessions. An SRC or an SRS SHOULD include the recording-session option tag in the Require header so that other types of user agents can simply reject the INVITE request with a 420 Bad Extension.

[5.3.](#) 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 4: 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.

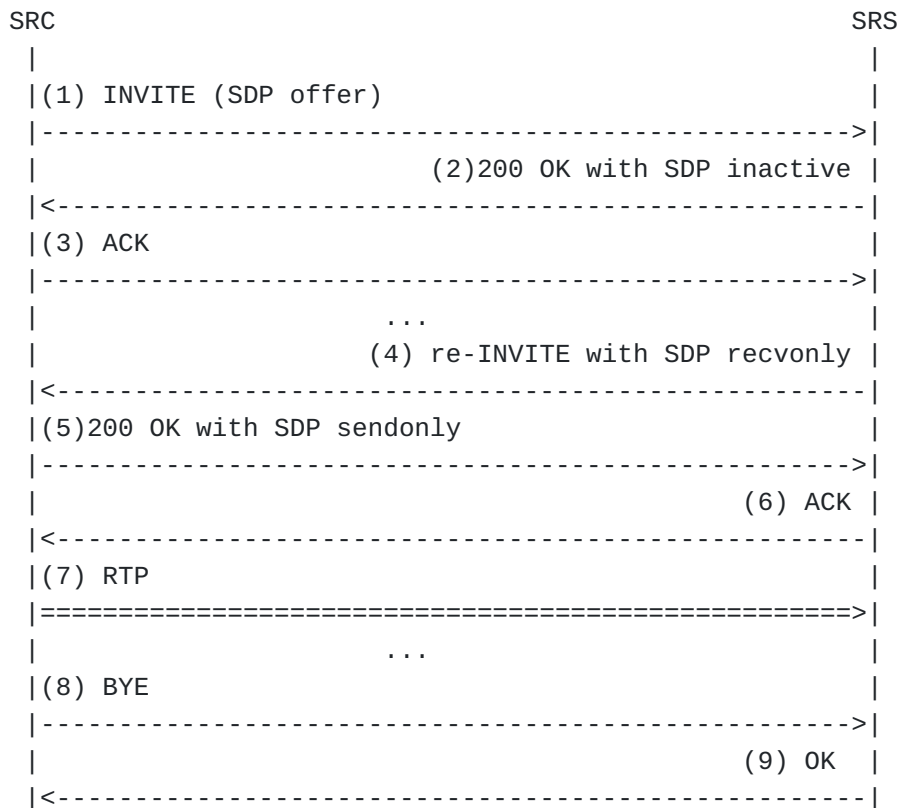


Figure 5: SRS to offer with a=inactive

5.4. 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 [\[I-D.eckel-siprec-rtp-rec\]](#)]

5.5. 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 describes the properties of media 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 and call identifiers, 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 is an 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
Max-Forwards: 70
Min-SE: 90
Session-Expires: 1800
Require: recording-session
Contact: <sip:2000@10.226.240.3:5060;transport=tcp>;src
Via: SIP/2.0/TCP 10.226.240.3:5060;branch=z9hG4bKdf6b622b648d9
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 6: Sample INVITE request for the recording session

Further updates to recording metadata can be delivered 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 can send a new metadata snapshot in a SIP reINVITE request along with an SDP offer. All subsequent metadata updates will be based on the new metadata snapshot.

5.6. Requesting for metadata snapshot

The SRS can 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 internal SRS error or failure to match metadata update sequence. Certain errors, such syntax errors or semantic errors in the metadata information, are likely caused by an error on the SRC side, and it is likely the same error will occur again when a new snapshot is requested. In order to avoid repeating the same error with snapshot requests, it is RECOMMENDED that the SRS terminate the recording session when a syntax error or semantic error occurs in the metadata. 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:

```
UPDATE sip:2000@10.226.240.3:5060 SIP/2.0
To: <sip:2000@10.226.240.3>;tag=35e195d2-947d-4585-946f-098392474
From: <sip:Recorder@10.240.3.10>;tag=1234567890
Call-ID: d253c800-b0d1ea39-4a7dd-3f0e20a@10.226.240.3
CSeq: 1 UPDATE
Supported: timer
Max-Forwards: 70
Min-SE: 90
Session-Expires: 1800
Require: recording-session
Contact: <sip:Recorder@10.240.3.10:5060>;srs
Via: SIP/2.0/UDP 10.240.3.10:5060;branch=z9hG4bKdf6b622b648d9
Content-Disposition: recording-session
Content-Type: application/rs-metadata-request
Content-Length: [length]

Reason: SRS internal error
```

Figure 7: Metadata Request

The SRS MAY include the reason why a metadata snapshot request is being made to the SRC in the Reason header. This header is free form text

mainly designed for logging purposes on the SRC side. 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. The processing of the content by the SRC is entirely optional since the content is for logging only, and the snapshot request itself is indicated by the use of the application/rs-metadata-request content type.

When the SRC receives the request for a metadata snapshot, the SRC MUST provide a metadata snapshot in a separate INVITE transaction, along with an SDP offer. All subsequent metadata updates sent by the SRC MUST be based on the new metadata snapshot.

5.6.1. 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-reason-line CRLF [ *opt-srs-headers ]
srs-reason-line  = "Reason" HCOLON srs-reason
srs-reason       = [TEXT-UTF8-TRIM]
opt-srs-headers  = CRLF 1*(extension-header CRLF)
```

5.7. Recording Pause and Resume

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

To resume streaming and collection of recorded media, the SRC sends 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 is 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. Extensions for Recording-aware User Agents

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

6.1. SIP Extensions

6.1.1. Recording awareness

A recording-aware UA SHOULD 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 automaton (eg. IVR, media server, PSTN gateway) may not have a user interface to render recording indication. When such user agent indicates recording awareness, the UA SHOULD 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 mechanism 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.

6.2. SDP Extensions

6.2.1. 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 in either SDP offer or answer. The recording indication applies to the specified media stream only, for example, only the audio portion of the call is recorded in an 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.

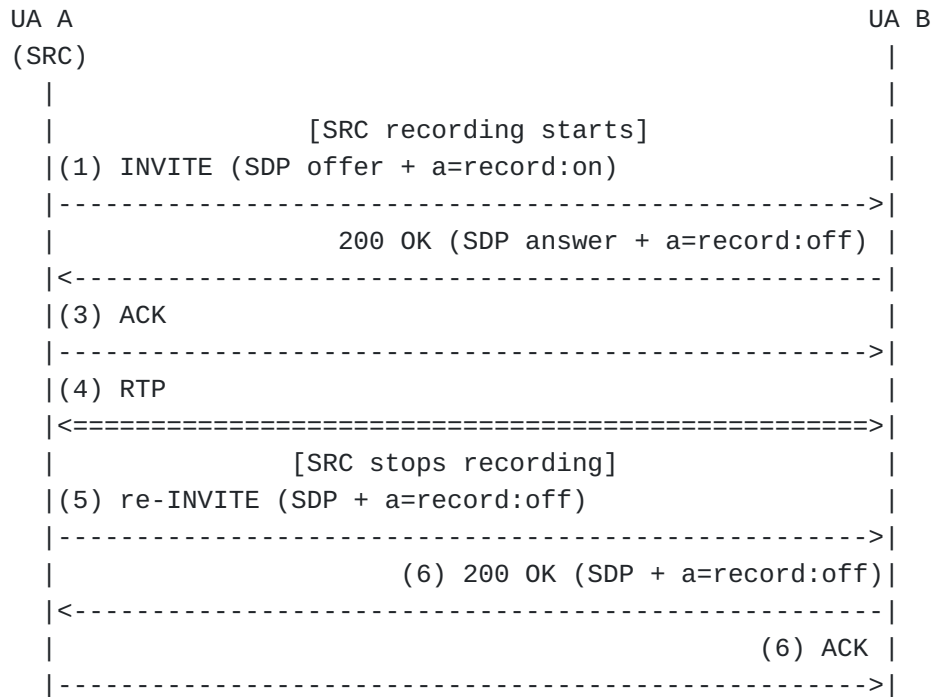


Figure 8: Recording indication example

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 and is not used as a request to start/stop recording if there are multiple SRCs in the call path.

6.2.2. 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 can 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. Registration of Option Tags

This specification registers two option tags. The required information for this registration, as specified in [\[RFC3261\]](#), is as follows.

7.1.1. recording-session Option Tag

*Name: recording-session

*Description: This option tag is for identifying the SIP session for the purpose of recording session only. This is typically not used in a Supported header. When present in a Require header in a request, it indicates that the UAS MUST be either a SRC or SRS capable of handling the contexts of a recording session.

7.1.2. 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.

7.2. Registration of media feature tags

This document registers two new media feature tags in the SIP tree per the process defined in [\[RFC2506\]](#) and [\[RFC3840\]](#)

7.2.1. src feature tag

*Media feature tag name: sip.src

*ASN.1 Identifier: 25

*Summary of the media feature indicated by this tag: This feature tag indicates that the user agent is a Session Recording Client for the purpose for Recording Session.

*Values appropriate for use with this feature tag: boolean

*The feature tag is intended primarily for use in the following applications, protocols, services, or negotiation mechanisms: This feature tag is only useful for a Recording Session.

*Examples of typical use: Routing the request to a Session Recording Server.

*Security Considerations: Security considerations for this media feature tag are discussed in Section 11.1 of RFC 3840.

7.2.2. srs feature tag

*Media feature tag name: sip.srs

*ASN.1 Identifier: 26

*Summary of the media feature indicated by this tag: This feature tag indicates that the user agent is a Session Recording Server for the purpose for Recording Session.

*Values appropriate for use with this feature tag: boolean

*The feature tag is intended primarily for use in the following applications, protocols, services, or negotiation mechanisms: This feature tag is only useful for a Recording Session.

*Examples of typical use: Routing the request to a Session Recording Client.

*Security Considerations: Security considerations for this media feature tag are discussed in Section 11.1 of RFC 3840.

7.3. 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

7.4. Media Type Registration

7.4.1. 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

7.4.2. 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

7.5. SDP Attributes

This document registers the following new SDP attributes.

7.5.1. '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

7.5.2. '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. Acknowledgements](#)

We want to thank John Elwell, Paul Kyzivat, Partharsarathi R, Ram Mohan R, Charles Eckel, Hadriel Kaplan, Adam Roach for their valuable comments and inputs to this document.

[10. References](#)

[10.1. Normative References](#)

[RFC2506]	Holtman, K., Mutz, A. and T. Hardie, "Media Feature Tag Registration Procedure" , BCP 31, RFC 2506, March 1999.
[RFC2119]	Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels" , BCP 14, RFC 2119, March 1997.
[RFC2234]	Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF" , RFC 2234, November 1997.
[RFC2804]	IABIESG, "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.
[RFC4574]	Levin, O. and G. Camarillo, "The Session Description Protocol (SDP) Label Attribute" , RFC 4574, August 2006.
[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.

[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 ", Internet-Draft draft-ietf-siprec-metadata-05, October 2011.

10.2. Informative References

[I-D.ietf-siprec-architecture]	Hutton, A, Portman, L, Jain, R and K Rehor, " An Architecture for Media Recording using the Session Initiation Protocol ", Internet-Draft draft-ietf-siprec-architecture-03, October 2011.
[I-D.eckel-siprec-rtp-rec]	Eckel, C, " Real-time Transport Protocol (RTP) Recommendations for SIPREC ", Internet-Draft draft-eckel-siprec-rtp-rec-03, October 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 Portman NICE Systems 8 Hapnina Ra'anana, 43017 Israel EMail: leon.portman@nice.com

Henry Lum editor Lum Genesys, Alcatel-Lucent 1380 Rodick Road, Suite 200 Markham, Ontario L3R4G5 Canada EMail: henry.lum@genesyslab.com

Alan Johnston Johnston Avaya St. Louis, MO 63124 EMail: alan.b.johnston@gmail.com

Andrew Hutton Hutton Siemens Enterprise Communications EMail: andrew.hutton@siemens-enterprise.com