Cisco BroadWorks

Call Recording Interface Guide

Release 23.0

Document Version 2

Notification

The BroadSoft BroadWorks has been renamed to Cisco BroadWorks. Beginning in September 2018, you will begin to see the Cisco name and company logo, along with the new product name on the software, documentation, and packaging. During this transition process, you may see both BroadSoft and Cisco brands and former product names. These products meet the same high standards and quality that both BroadSoft and Cisco are known for in the industry.

Copyright Notice

Trademarks

Any product names mentioned in this document may be trademarks or registered trademarks of Cisco or their respective companies and are hereby acknowledged.

Document Revision History

Release	Version	Reason for Change	Date	Author
18.0	1	Created initial document.	August 26, 2011	Carl Droste
18.0	1	Updated document.	September 13, 2011	Scott Orton
18.0	1	Updated document with review comments.	September 16, 2011	Scott Orton, Margaret Henderson
18.0	1	Edited and published document.	September 23, 2011	Jessica Boyle
19.0	1	Updated document for Release 19.0.	September 27, 2012	Carl Droste
19.0	1	Edited changes and published document.	November 12, 2012	Jessica Boyle
20.0	1	Updated document for Release 20.0.	October 10, 2013	Carl Droste
20.0	1	Edited changes and published document.	November 6, 2013	Joan Renaud
21.0	1	Updated document for Release 21.0.	October 21, 2014	Carl Droste
21.0	1	Updated the legal notice and edited changes.	October 30, 2014	Joan Renaud
21.0	1	Rebranded document.	November 5, 2014	Joan Renaud
21.0	1	Added limitation for Silent Monitoring.	November 5, 2014	Carl Droste
21.0	1	Edited changes and published document.	November 5, 2014	Joan Renaud
21.0	2	Updated server and client icons and published document.	February 26, 2015	Joan Renaud
21.0	3	Updated section *4.3.4 Redirected Calls* for behavior change on the Auto Attendant call for PR-46924.	March 24, 2015	Xiufen Deng
21.0	3	Edited changes and published document.	June 3, 2015	Joan Renaud
21.0	4	Added SIP compliance to *RFC 3311* to section *4.2 SIP Compliance* for PR-47430.	August 6, 2015	Carl Droste
21.0	4	Edited changes and published document.	November 4, 2015	Joan Renaud
22.0	1	Updated document for Release 22.0.	September 26, 2016	Carl Droste, Scott Orton
22.0	1	Edited changes and published document.	November 24, 2016	Joan Renaud
23.0	1	Updated document for Release 23.0. Added section *5.13.14 Emergency Calls* for PR-54098. Updated sections *4.3.2 Originating Calls* and *4.3.3 Terminating Calls* to clarify connected and Calling Line ID privacy. Added section *5.8 In-band versus Out-of-band DTMF Transmission*. Updated section *6.3.5 Add Call Recording Platform(s)*, which describes the FQDN resolution and how the request URI to the third-party call recording platform is formatted.	September 17, 2018	Carl Droste, Scott Orton

Release	Version	Reason for Change	Date	Author
23.0	1	Edited changes and published document.	September 18, 2018	Patricia Renaud
23.0	2	Updated document to include changes introduced in FR17535 Container Option Conversions – Call recording, Push Notification, Conference, and Meet-Me.	February 21, 2019	Scott Orton
23.0	2	Rebranded product name for Cisco. Edited changes and published document.	March 7, 2019	Joan Renaud

Table of Contents

Table of Figures

1 Summary of Changes

This section describes the changes to this document for each release and document version.

1.1 Changes for Release 23.0, Document Version 2

This version of the document includes the following changes:

- Updated document to include changes introduced in FR17535 Container Option Conversions – Call recording, Push Notification, Conference, and Meet-Me.
- Rebranded product name for Cisco.

1.2 Changes for Release 23.0, Document Version 1

This version of the document includes the following changes:

- Changed document revision to Release 23.0.
- Added section *5.13.14 Emergency Calls*.
- Updated sections *4.3.2 Originating Calls* and *4.3.3 Terminating Calls* to clarify connected and calling line ID privacy.
- Added section *5.8 In-band versus Out-of-band DTMF Transmission*.
- Updated section *6.3.5 Add Call Recording Platform(s)* concerning the fully qualified domain name (FQDN) resolution and how the request URI to the third-party call recording platform is formatted.

1.3 Changes for Release 22.0, Document Version 1

This version of the document includes the following changes:

- Changed document revision to Release 22.0.
- Updated document for FR6899: SRV Support for Call Recording Platforms. This feature adds the ability to support Service Locator (SRV) records for the FQDN address resolution of the Call Recording platforms.
- Removed items from section *6.1.1 System Level*. For more information, see section *6.3.6.2 Parameters Removed Due to SRV Support*.

1.4 Changes for Release 21.0, Document Version 4

This version of the document includes the following change:

- Added SIP compliance to *RFC 3311* in section *4.2 SIP Compliance* for PR-47430.

1.5 Changes for Release 21.0, Document Version 3

This version of the document includes the following change:

- Updated section *4.3.4 Redirected Calls* for behavior change on the Auto Attendant call for PR-46924.

1.6 Changes for Release 21.0, Document Version 2

This version of the document includes the following change:

- Updated server and client icons.

1.7 Changes for Release 21.0, Document Version 1

This version of the document includes the following changes:

- Call Recording Manager now supports 302 redirection for scalability and load balancing.
- Media Server redundancy support for Call Recording – An alternate Media Server can be leveraged to provide failure recovery to a recorded call.
- Added a limitation for Silent Monitoring.

1.8 Changes for Release 20.0, Document Version 1

Several features have been added to Call Recording for Release 20.0:

- FR170972: Start, Stop, Pause, Resume – This capability is added for user control of Call Recording on a current call. See the *Call Recording Start Stop Pause Resume User Control Feature Description* [12] for this feature.
- FR170973: Video Support – Provides the capability to record video in addition to audio. It interfaces with the Third Party Call Recording platform (3PCR) using SIP to support audio and video recordings in both single and dual modes. The Cisco BroadWorks Media Server is used to stream media to both the users and 3PCR platform. See the *Call Recording Video Support Feature Description* [14] for this feature.
- FR170971: End-User Notification – Adds the capability to notify callers when a call is being recorded. See the *Call Recording End-User Notification of Recording Feature Description* [13] for this feature.
- FR170964: Voice Mail Recording – Enhances the existing Call Recording service by adding the capability to record both the voice mail messages and the caller's interactions with the Voice Messaging system while callers are leaving messages in the user's mailbox. See the *Call Recording Voice Mail Recording Feature Description* [15] for this feature.
- FR170963: Control for IP Phones – Provides the additional interface between the Cisco BroadWorks Application Server (AS) and SIP endpoint devices, which allows users to initiate and control their call recording sessions from their IP phones. See the *Call Recording Controls for IP Phones Feature Description* [16] for this feature.
- FR170975: Media Server Optimized Port Usage – Optimizes the generic port resource locking for audio performed by the Cisco BroadWorks Media Server when servicing *msc-mixer* join requests between connections established via the Cr Interface's *cfw-media* SIP Media Dialogs, as occurs in the case of Call Recording. The optimization reflects the lower generic port resource requirements for joins that do not require audio transcoding. See the *Media Server Performance Optimizations for Call Recording Port Usage Feature Description* [18] for this feature.
- FR170974: Visual Security Classification for Active Calls – Metadata support added to Call Recording for this security feature which introduces the Security Classification service that allows Cisco BroadWorks to classify calls in which a user is involved, with a security classification level. See the *Visual Security Classification for Active Call Feature Description* [17] for this feature.

1.9 Changes for Release 19.0, Document Version 1

This version of the document includes the following changes:

- Updated additional Metadata information.
- Added additional service interactions to section *5 Call Recording Feature Details and Service Interactions*.
- Added additional limitation with *SessionAudit*.

1.10 Changes for Release 18.0, Document Version 1

This is the initial release of this document.

2 Introduction

This document describes the configuration and details of the Cisco BroadWorks Call Recording Interface and the Call Recording service.

This interface allows a Cisco BroadWorks deployment to record calls onto a Third Party Call Recording (3PCR) Platform. The interface is based on the work being done by the IETF SIPREC working group. The third-party platform is responsible for the media storage and management of the archived call recordings. Cisco BroadWorks only provides the media streams of the call and information about the parties in the call.

The Call Recording service on the Cisco BroadWorks Application Server is a user assignable service that allows a user to record calls onto the configured third-party platform. This service can be assigned to Cisco BroadWorks users and the hosted Private Branch Exchange (PBX) and Business Connectivity applications, in particular:

- Call Center
- Route Point
- Attendant Console

The Call Recording service has been expanded to support:

- Start, Stop, Pause, and Resume
- Video
- End-User Notification of Recording
- Voice Mail Recording
- User Control for IP Phones
- Visual Security Classification for Active Call

3 Cisco BroadWorks Call Recording Interface

The Cisco BroadWorks Call Recording interface requires the following components:

- Cisco BroadWorks Application Server
- Cisco BroadWorks Media Server
- Third Party Call Recording Platform

The following figure shows the connections and protocols used to connect the components. It shows the Xtended Services Platform (Xsp), clients, and 3PCR portal, which are not required components to successfully record a call. The 3PCR portal provides enhanced capabilities to manage the recordings and to specify that an *On-Demand* recording be kept.

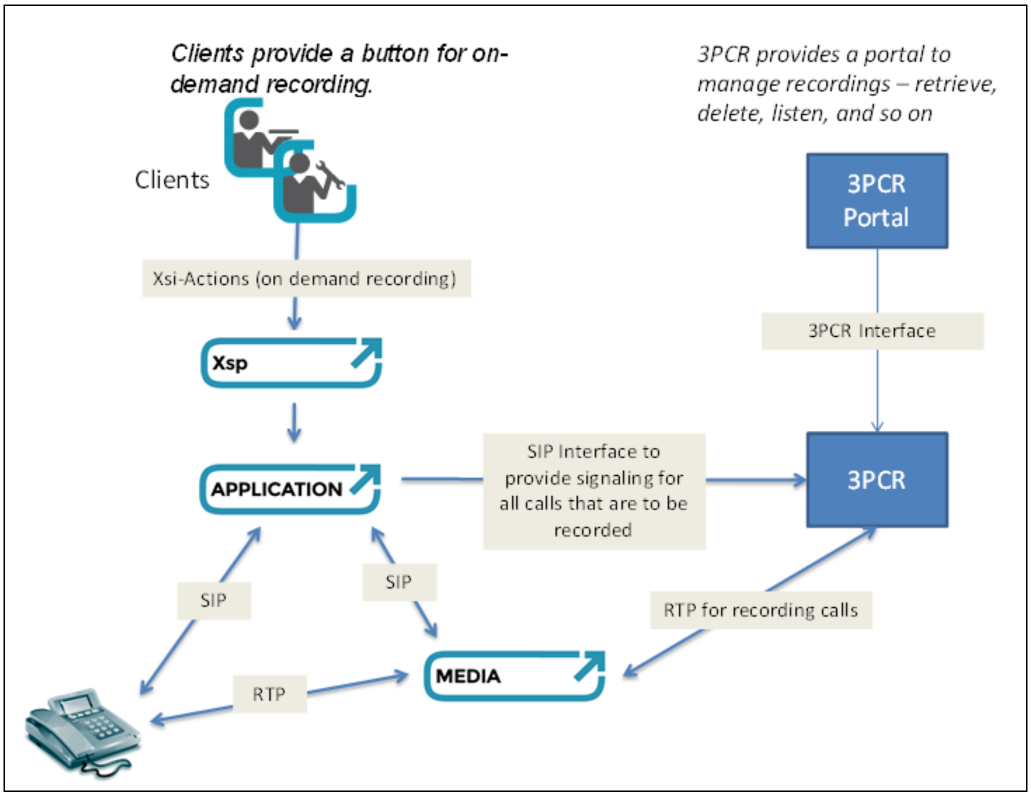

Figure 1 Component Interconnection

3.1 Roles of Cisco BroadWorks Elements

Cisco BroadWorks Call Recording relies on the interactions of several Cisco BroadWorks elements as described in the following sections.

3.1.1 Cisco BroadWorks Application Server Role

The Application Server is the primary component of the Cisco BroadWorks Call Recording feature. It coordinates the connections between the clients generating the Real-Time Transport Protocol (RTP) streams being recorded, the Cisco BroadWorks Media Server and the Third Party Call Recording Platform.

The Call Recording service creates connections on the Cisco BroadWorks Media Server for the clients that are being recorded and for the connections to the 3PCR platform. It controls how the connections are mixed to ensure that the clients receive the correct media streams and that the 3PCR platform only has a one-way voice path to receive the RTP streams from the Media Server.

Additionally, the Call Recording service creates a connection to the 3PCR platform. The connections to the 3PCR platform are managed through SIP. The INVITE messages that create the connections contain a multipart MIME body. The body contains eXtensible Markup Language (XML) metadata that provides information about the call being recorded in addition to a Session Description Protocol (SDP) description of the media connections for the RTP stream(s).

The media sent to the 3PCR platform can be configured so that it is in a single stream (single) or in two separate streams (dual). The format is configured when the 3PCR platform is provisioned on the Cisco BroadWorks system.

For Audio

In the case where there are two streams, the audio *from* the user who is recording the call is in one stream, and the other stream contains the audio *sent to* the user.

For Video

The video sent to the 3PCR platform uses the existing provisioning for the streaming mode and is sent in a single stream (single) or in two separate streams (dual). The mode is configured when the platform is provisioned on the Cisco BroadWorks system. In dual mode there are two streams, the video *from* the user who is recording the call is in one stream and the video *received* by the user is in the other stream.

A user's Call Recording service can be configured in four different modes: *Always, Always with Pause/Resume, On-Demand,* and *On-Demand with User Initiated Start*.

When *Always* is selected, all of the calls a user makes or receives are automatically recorded.

When Always with Pause/Resume is selected the recording of the call can be paused and then resumed for the same call recording.

In *On-Demand* mode, the call is recorded from the start, but the 3PCR platform discards the recording at the end of the call unless informed by the Application Server to keep the call. The request to keep an *On-Demand* recording can be made either through a feature access code (*44) or an Xtended Services Interface (Xsi) request. The request to keep the call must occur prior to the call ending. The feature access code (FAC) code can be input prior to the call or during the call. The Xsi request is supported mid-call only. When the FAC or Xsi request is received mid-call, an INFO message is sent to the 3PCR platform to request that it keep the recording for the call. The service can also be disabled by setting the mode to *Never*. The *On-Demand* mode is not available if the user is a virtual user, such as a Call Center.

The existing *On Demand* mode is enhanced to support "pause" and "resume". In addition, a new system parameter is introduced, which controls whether the call continues or fails (if there is a failure setting up the recording or in recording the call when one of the two *Always* modes is selected). The following sections provide further information regarding the enhancements.

This Start, Stop, Pause, Resume feature also enhances the Call Center and Receptionist applications to provide the user with controls to start, stop, pause, and resume call recordings. These controls are applicable to the new recording modes as previously mentioned.

3.1.2 Cisco BroadWorks Media Server Role

The Cisco BroadWorks Media Server provides the media anchor between the clients, and it redirects a copy of the RTP streams to the 3PCR platform. The Media Server is supervised by the Cisco BroadWorks Application Server to create the connections and the copy of the RTP streams.

3.1.2.1 Video Optimization Support

Release 20.0 adds optimizations for generic port resource locking for audio performed by the Cisco BroadWorks Media Server when servicing *msc-mixer* join requests between connections established via the Cr Interface's *cfw-media* SIP Media Dialogs, as occurs for Call Recording. The optimization reflects the lower generic port resource requirements for joins that do not require audio transcoding.

For more information, see the *Media Server – Performance Optimizations For Call Recording Port Usage Feature Description* [22].

3.1.3 Clients Role

The clients play a role in recording when the call is being recorded in *On-Demand* and *On Demand with User Initiated Start* modes.

In *On-Demand*, the call is recorded from the beginning, but the 3PCR platform discards the recording at the end of the call unless informed by the Application Server to keep the call. The client can inform the Application Server of its desire to keep the recording by either entering the FAC code *44, or if it is capable, by sending an Xsi request.

The *On Demand with User Initiated Start* mode can initiate a call recording by using the *44 FAC followed by a destination number, the *44 FAC by itself, or by sending in the existing Xtended Services Interface record command.

3.1.3.1 Start, Stop

Release 20.0 introduced new functionality that allows a user with the *On Demand with User Initiated Start* mode to start recording a call at a time of their choosing instead of automatically at the beginning of the call, as is done for the *On Demand* mode. In addition, the user can stop the call recording at any time, which ends the recording and the connection to the 3PCR platform. If the user starts to record the call after stopping a recording, a new call recording is generated.

Users with *On Demand* mode or *On Demand with User Initiated Start* mode can initiate a call recording by using the *44 FAC followed by a destination number, the *44 FAC by itself, or by sending in the existing Xtended Services Interface (Xsi) record command.

Stopping a call recording can be initiated by using the new *45 FAC or by sending in the new Xtended Services Interface *stopRecording* command.

3.1.3.2 Pause/Resume

Release 20.0 introduced the pause/resume functionality that allows a user to pause the recording of the call and then to resume the same call recording. The pause functionality is available under one of the following conditions:

- The user's call recording mode is *Always with Pause/Resume.*
- The user's call recording mode is *On Demand*, after the user has decided to keep the call recording using either the *44 FAC or the Xtended Services Interface *record* action.
- The user's call recording mode is *On Demand with User Initiated Start*, after the user has started recording the call using either the *44 FAC or the Xtended Services Interface record action.

A call can be paused using either the *48 FAC or an Xsi request. The participants of the call can be informed of the call recording being paused by playing a tone or an announcement to all participants of the call. Pausing can also be done with no notification to the parties in the call.

A call can be resumed only after being paused, by using either the *49 FAC or the Xtended Services Interface *resumeRecording* request. The participants of the call can be notified by an announcement or tone that the call recording has resumed, as previously described.

3.1.3.3 IP Phone Client Controls

Release 20.0 provided a client interface between the Cisco BroadWorks Application Server (AS) and SIP endpoint devices, which allows users to initiate and control their call recording sessions from their IP phones.

3.1.4 Cisco BroadWorks Supported 3PCR Server Role

The 3PCR platforms are provisioned on the Cisco BroadWorks system at the system level, and multiple 3PCR platforms can be defined. The system administrator specifies which 3PCR platforms is a default. The system administrator also specifies which recording platform to use at the group level if a group does not want to use the default platform.

The 3PCR server is responsible for storing the call recordings that are requested, and it is responsible for providing a mechanism to retrieve and playback the recordings.

Additional information on the role of the 3PCR platform is detailed in section *4 Call Recording Platform Interface*.

4 Call Recording Platform Interface

The 3PCR platform is responsible for recording the media that is streamed from the Cisco BroadWorks platform. The Cisco BroadWorks Application Server sets up the connection to the 3PCR system using SIP; as such, the 3PCR platform is expected to perform primarily as a SIP user agent (UA), defined by *RFC 3261*.

The 3PCR platform is also expected to support the Cisco BroadWorks SIP INFO event package, *x-broadworks-callrecording,* for the purpose of informing the 3PCR platform that an *On-Demand* recording should be kept.

The Cisco BroadWorks Application Server sends information in the message body of the INVITE in the form of XML-based metadata during the setup of the call recording session. The 3PCR platform should store and be able to retrieve call recordings based on the metadata that the Cisco BroadWorks Application Server provides. The information of specific interest is contained in the extension data of the communication session defined in *Session Initiation Protocol Recording Metadata Format* [2].

Since the media connection for call recording is one-way, the 3PCR platform is expected to use the *recvonly* attribute in all of its SDP descriptions. The Cisco BroadWorks Application Server sends the *sendonly* attribute to the 3PCR platform. The only exception to these rules is if the recording needs to be paused, in which case, the Application Server sends *inactive,* and it is expected that the 3PCR platform will send the same *inactive* attribute in the response. When the recording resumes, the Application Server sends *sendonly* and expects the 3PCR platform to respond with *recvonly*. For more information on the messaging between the two components, see the examples in section *10 Call Flow Details*.

4.1 Release 20.0 Changes

The Call Recording of Voice Mail Recording [15] and the Visual Security Classification for Active Call [17] features are the only ones that introduce any changes to the metadata definition. The call flows are modified. Call recording does not start until the call is answered. Waiting until the call is answered does not change the message flows to the third-party call recording platforms; it only delays when the exchange starts with respect to the call start. However, these features do change the interactions with the 3PCR platforms when SDP is involved in the exchange.

4.1.1 SDP Exchanges

Prior to Release 20.0, all SDP negotiations after the initial setup of the session were handled using the SIP UPDATE method. To help with interoperability issues, these interactions are now performed using the INVITE method. The UPDATE method is still used for changes that only affect the metadata.

4.1.2 Start/Stop

Starting or stopping a call recording in the middle of a call does not change from the existing call flows to the 3PCR platform. The only difference is when recording on a conference call, the initial metadata contains more than two participants. Prior to this feature, all call recordings started with only two participants.

4.1.3 Pause/Resume

To notify the 3PCR platform that a recording has been placed on hold, the Application Server sends an INVITE with all of the media streams marked as *inactive*. When the recording is resumed, an INVITE is sent to 3PCR platform with all of the media streams marked as *sendonly*, *sendrecv*, or as a no direction attribute, which should be interpreted as *sendrecv*.

4.1.4 Multiple Recordings for Same Call

The one change that needs to be handled on the third-party call recording platform(s) is the new ability to have multiple call recordings for the same call. The information in the metadata for multiple recordings for the same call only differs in the start times and the universally unique identifiers (UUIDs). Therefore, care must be taken in storing the recordings for later retrieval.

4.1.5 Video Support Changes

The interface to the 3PCR platform now includes the video SDP information and follows the same format as that used for the audio SDP. Each video stream in the SDP has an associated label similar to the audio stream. The Cisco BroadWorks Telephony Application Server sends information in the message body of the INVITE in the form of XML-based metadata during the setup of the call recording session. Video SDP stream information is added to the metadata for each participant in the call being recorded. The metadata format is the same as that used for the audio streams.

Since the media connection for call recording is one way, the 3PCR platform is expected to use the *recvonly* attribute for the video SDP as well, in all of its SDP descriptions. The Cisco BroadWorks Telephony Application Server sends the *sendonly* attribute to the 3PCR platform. The only exception to these rules is if the recording needs to be paused, in which case, the Cisco BroadWorks Telephony Application Server sends *inactive,* and it is expected that the 3PCR platform should send the same *inactive* attribute in the response. When the recording resumes, the Cisco BroadWorks Telephony Application Server sends *sendonly* and expects the 3PCR to respond with *recvonly*.

4.2 SIP Compliance

The 3PCR platform should comply with the following Request for Comments (RFCs):

- *RFC 3261* [3]
- *RFC 4574* [4]
- *RFC 6086* [5]
- *RFC 3311* [25]

In addition, parts of the following draft specifications are supported. The following subsections provide additional details on the compliance with these draft specifications:

- *draft-portman-siprec-protocol-03* [1]
- *draft-ram-siprec-metadata-format-01* [2]
- *draft-ietf-siprec-protocol-09* [19], sections *7.1.2, 7.3.2, 11.1.2*, and *11.5*

4.2.1 RFC 4574 − SDP Label Attribute

To identify whether the Cisco BroadWorks Application Server is setting up a dual/single RTP media stream, the SDP's *label* attribute is used. Each media description sent by the Cisco BroadWorks Application Server is identified by a label as described in *RFC 4574*. The 3PCR platform must accept and start recordings for each of the media streams. The 3PCR system should use the information in the metadata to index and store these recordings.

The SDP responses sent by the 3PCR platform must identify the media descriptions using the same labels. Additional information on the usage of labels can be found in section *5.2* of *draft-portman-siprec-protocol-03* [1]. Note that the Cisco BroadWorks implementation only sends at most two streams to the Call Recording Server. If there is a single stream, then all of the participants are mixed into the single stream. If a dual stream is provided, then one stream contains the audio from the user recording the call and the other stream contains the audio that is sent to the user recording the call.

4.2.1.1 Video Support

Each media description sent by the Cisco BroadWorks Telephony Application Server is identified by a label as described in *RFC 4574*. The 3PCR platform must accept and start recordings for each of the media streams. The 3PCR system should use the information in the metadata to index and store these recordings.

In addition, the SDP responses sent by the 3PCR platform must identify the media descriptions using the same labels. Additional information on the usage of the labels can be found in section *5.2* of *draft-portman-siprec-protocol-03* [1]. Note that the Cisco BroadWorks implementation sends at most four streams to the Call Recording platform when audio and video are being recorded.

- If in single mode, then two streams are sent, one for audio and one for video. Each of the streams have all of the participants mixed into a single stream for audio and a single stream for video.
- If in dual mode, then four streams are sent. One stream contains the audio and another stream contains the video *from* the user recording the call. In addition, another stream contains the audio, with another video that is *sent* to the user recording the call.

For example messages, see sections *10.1.1 INVITE to Call Recording Platform, 10.1.2 200 OK from Call Recording Platform*, *10.5.1 INVITE to Call Recording Platform*, and *10.5.2 200 OK from Call Recording Platform*.

4.2.2 RFC 6086 − Session Initiation Protocol INFO Method and Package Framework

This RFC defines the ability to include event packages in the SIP INFO method. It adds two new headers to the SIP messages. Call Recording makes use of these new headers and defines a new INFO event package named *x-broadworks-callrecording.* This new event package is used for signaling that an *On-Demand* recording should be kept by the 3PCR platform.

- *Info-Package* – This header specifies which information event package the INFO method contains. The Call Recording service sends the *x-broadworks-callrecording* package to indicate that the 3PCR platform should save the recording belonging to the SIP dialog for which this INFO message belongs.

- *Recv-Info* – This header specifies what Info packages are supported by a client or a server. The 3PCR platform must send the *Recv-Info* with the *x-broadworks-callrecording* event package in the *200 OK* response to the INVITE to inform the Cisco BroadWorks Application Server that it supports *On-Demand* recording. In addition, an empty *Recv-Info* header is sent in the initial INVITE by the Application Server to let the 3PCR platform know that the Application Server supports the INFO event package.

For information on the new INFO event package, see section *9.2.4 Info-Package*.

For example, messages, see sections *10.1.2 200 OK from Call Recording Platform*, *10.5.2 200 OK from Call Recording Platform*, and *10.6.1 INFO Message*.

4.2.3 draft-portman-siprec-protocol-03

This section highlights the portions of the protocol draft that the Cisco BroadWorks Application Server sends or expects to receive from the 3PCR platform. The Cisco BroadWorks Application Server does not support the following sections of the *draft-portman-siprec-protocol-03*:

- Section 6 SIP Extensions for Recording-Aware Devices
- The new mime type application/rs-metadata-request in section 7.2.2
- The new Info-Package recording-session-srs

4.2.3.1 Contact Parameters

This draft defines two new *Contact* header parameters, which are used by the parties in a call recording to signify the roles that the parties identified by the *Contact* headers are playing. The 3PCR platform should include the *srs* parameter as defined in section 5.1.2 of the draft specification. The Cisco BroadWorks Application Server includes the *src* contact parameter as defined in section 5.1.1 of the draft specification.

- Support the following *Contact* header parameters:
  - *src* – This is sent by the Cisco BroadWorks Application Server in the *Contact* header.
  - *srs* – This is sent by the 3PCR platform in the *Contact* header.

For examples of messages using this contact parameter, see section *10.1.1 INVITE to Call Recording Platform.*

4.2.3.2 Content-Type

The new *application/rs-metadata* content-type is sent to the 3PCR platform when the metadata is present in the message body.

4.2.3.3 Content-Disposition

When the metadata is present in the message body, the new *recording-session* content-disposition type is also sent to the 3PCR platform.

4.2.4 draft-ram-siprec-metadata-format-01

The metadata format is adhered to in its entirety. This is reflected in the XML schema defined in section *9.6.1 rs-metadata+xml*.

The Call Recording service also defines Cisco BroadWorks-specific extension data to be included as part of the extension data for the communication session. For information on the extension data, see sections *4.3 Cisco BroadWorks-specific Extension Data* and *9.6.2 broadWorksRecordingMetadata*.

4.2.5 draft-ietf-siprec-protocol-09, Sections 7.1.2, 7.3.2, 11.1.2, and 11.5

The Call Recording Controls for IP Phones feature only adds support for the recording-aware UA. It does not update the interface between the Cisco BroadWorks Application Server and the 3PCR platform.

4.2.5.1 Option Tag

The Cisco BroadWorks Application Server supports a new option tag, *record-aware*, which can be present in the *Supported* header or in the *Require* header of a SIP message. The presence of this tag means that the recording-aware UA can send and receive the SDP attributes, *recordpref* and *record*, to request the recording preferences, and to obtain the current call recording state.

When the Cisco BroadWorks Application Server receives the *record-aware* option tag from the device, the Cisco BroadWorks Application Server reports the current recording status of the call for the user who sent the *record-aware* option tag.

For more information on the *record-aware* option tag, see the *Call Recording Controls for IP Phones Feature Description* [16].

4.2.5.2 SDP Attributes

The Cisco BroadWorks Application Server supports two new SDP attributes, *recordpref* and *record*, which can be present at the session level of the SDP or at the media level.

For more information about the *recordpref* and *record* attributes, see the *Call Recording Controls for IP Phones Feature Description* [16].

4.2.5.2.1 recordpref

The *recordpref* attribute is sent by the recording-aware UA. This attribute indicates how the recording-aware UA prefers the Cisco BroadWorks Application Server to handle the recording of the call, and it can be sent in the SDP of an INVITE request, INVITE response, or UPDATE request.

- By setting the recording preference to “on”, the recording-aware UA wants to start a recording or to resume the paused recording.
- By setting the recording preference to “off”, the recording-aware UA wants to stop the existing recording.
- By setting the recording preference to “pause”, the recording-aware UA wants to pause a recording that is currently in progress.
- By setting the recording preference to “nopreference”, the recording-aware UA has no preference on recording.

The *recordpref* attribute is also accepted by the Cisco BroadWorks Application Server if the recording-aware UA sends it in the initial INVITE during call setup. It provides *On Demand* and *On Demand with User Initiated Start* users with the ability to start the recording while the call is being established, instead of in the middle of the call.

4.2.5.2.2 record

The *record* attribute is sent by the Cisco BroadWorks Application Server in the SDP of an INVITE request, an INVITE response, or an INVITE response acknowledgement. This attribute reports the state of the recording to the recording-aware UA, since Cisco BroadWorks is recording the call on behalf of the user.

- When it is “on”, the recording is in progress.
- When it is “off”, there is no recording in progress.
- When it is “paused”, the recording is in progress but the media is paused.

For multiple recording-aware devices involved in the same call, the Cisco BroadWorks Application Server maintains the recording state of calls on a half-call basis. In a two-party call between User A and User B, each party may start their own recording of the call. However, the information reported to User A in the *a=record* SDP attribute only applies to the recording being performed on behalf of User A. Similarly, the *a=record* attribute reported to User B only applies to the recording being performed on behalf of User B.

4.2.5.3 Recording-Aware UA Expectations

A few specific expectations are highlighted in the following subsections.

4.2.5.3.1 recordpref

If the recording-aware UA sends the *a=recordpref* attribute with a value that is not valid for the user’s provisioned recording mode, then the recording preference is ignored on the Cisco BroadWorks Application Server. For example, if the user is assigned the *Always* recording mode and if the device allows the user to pause the recording, the recording preference to pause is not allowed in the *Always* recording mode and it is ignored on the Cisco BroadWorks Application Server.

If the recording-aware UA sends the *a=recordpref* attribute with a value that results in an action that is a duplicate to the previous request, then the recording preference is ignored at the Cisco BroadWorks Application Server, and the state of the recording remains unchanged. For example, if the user is assigned the *Always with Pause/Resume Support* recording mode and if the device allows the user to pause the recording multiple times in a row, then the subsequent recording preferences to pause are ignored because the call is already paused.

4.2.5.3.2 record

When indicating recording-awareness in the *Supported* header with the *record-aware* option tag, if the Cisco BroadWorks Application Server does not explicitly indicate that the call recording is “on” or “off” in the SDP’s *a=record* attribute, it is safe for the UA to assume the recording is “off”, until the Cisco BroadWorks Application Server indicates otherwise.

4.2.5.3.3 UPDATE

When the recording-aware UA requests to change the recording by using the *a=recordpref* SDP attribute, the UA may choose to send the SDP attribute in a re-INVITE or an UPDATE request to the Cisco BroadWorks Application Server. The Cisco BroadWorks Application Server is able to accept both.

Even if the recording-aware UA indicates that it understands receiving UPDATEs, the Cisco BroadWorks Application Server always sends a re-INVITE if it must convey the *a=record* SDP attribute in a SIP request. By always sending a re-INVITE, it increases the Cisco BroadWorks Application Server interoperability, since some UAs indicate they support receiving UPDATE requests but cannot actually support SDP negotiation using UPDATEs.

4.3 Cisco BroadWorks-specific Extension Data

The Call Recording service defines a new XML schema for the data that the Cisco BroadWorks Application Server associates with the call recording. The 3PCR platform needs to be able to parse and use this data to store and retrieve the call recording. The XML schema for this extension data is defined in section *9.6.2 broadWorksRecordingMetadata*. This extension data contains the following:

Identifier	Values	Description
extTrackingID	string	This is the identifier for the recording used to identify all recordings. It can span multiple recordings in cases where the call is transferred, to group multiple recordings together.
serviceProviderID	string	This is the identifier for the name of the Cisco BroadWorks service provider or enterprise to which the user requesting the recording belongs.
groupID	string	This is the identifier for the group to which the user requesting the recording belongs.
userID	string	This is the identifier for the Cisco BroadWorks user for whom the call is being recorded.
callID	string	This is the SIP call ID for the call being recorded.
callType	“origCall” or “termCall”	This identifies how the call was set up. ▪ “origCall” specifies the user placed the call. ▪ “termCall” specifies the user received the call.
recordingType	“on”, “demand”, or “voicemail”	This identifies the type of recording. For both types, the 3PCR platform should start recording the call. ▪ If *on* is specified, the 3PCR platform should keep the recording. ▪ If *demand* is specified, the 3PCR platform only keeps the recording if an INFO message is received with the *info-package* header set to “x-broadworks-callrecording”. ▪ If voicemail is specified, the 3PCR platform keeps the recording, but the contents are those of a voice message deposit being left for the user.
Acd	“acdUserId” “acdName” “acdNumber”	This identifies information about the call center being used to terminate to the agent. ▪ “acdUserId” – User Id of the call center/route point being used to reach the agent. ▪ “acdName” – The name configured against the call center/route point. ▪ “acdNumber” – The address for the call center/route point used to reach the agent. For example, the primary directory number (DN) or a Dialed Number Identification Service (DNIS) configured against the call center/route point.
callClassification	string	This is the highest security classification level reached for the call.

4.3.1 Call Center/Route Point Calls

If the call center/route point call is redirected to an agent, then additional information is captured in the *acd* element of the Cisco BroadWorks-specific extension metadata. The *acd* element is of type *acdDetails*, and it contains the following elements:

Identifier	Values	Description
acdUserId	String	This is the user ID of the call center/route point being used to reach the agent.
acdName	String	This is the name configured against the call center/route point.
acdNumber	acdNumberDetails	This is the address being used to reach the agent. This can be a primary DN or a DNIS configured against the call center/route point. If available, the country code is indicated as well.

4.3.2 Originating Calls

If the call type of the recording is *origCall*, then additional information is included as shown in the following table.

Identifier	Values	Description
callingPartyNumber	string	This is the phone number or extension that identifies the calling party.
calledPartyNumber	string	This is the called party number after translations.
dialedDigits	string	This is the digits dialed by the subscriber.

The 3PCR platform is considered an untrusted node, so the format of the *calledPartyNumber* honors the connect line ID privacy settings when sending metadata to the 3PCR platform.

4.3.3 Terminating Calls

If the call type of the recording is *termCall,* then additional information is included as shown in the following table.

Identifier	Values	Description
callingPartyNumber	string	This is the phone number or extension that identifies the calling party.
calledPartyNumber	string	This is the called party number after translations.
redirectedFromInfo	string	This is the last redirection number, original called number, and number of redirections if available. This is only when the call was redirected to the party recording the call.

The 3PCR platform is considered an untrusted node so the format of the *callingPartyNumber* honors the Calling Line ID privacy settings when sending metadata to the 3PCR platform.

4.3.4 Redirected Calls

This data is optional and is sent to the 3PCR platform when a user redirects a call after answer. The information in the following table is included for redirected calls.

Identifier	Values	Description
newExtTrackingID	string	This is the identifier for the new external tracking ID. In some scenarios when the call is redirected, the external tracking ID changes. This reports the new value.
redirectedFromPartyNumber	string	This is the phone number or extension that identifies the redirected *from* party.
redirectedToPartyNumber	string	This is the phone number or extension that identifies the redirected *to* party.

The *newExtTrackingID* field is not shown in some redirected call scenarios, such as the Auto Attendant call. In those cases, after the call is redirected, the existing connection to the call recording platform is torn down and a new connection is established after the redirecting destination answers.

4.4 Fast Answer Support

To minimize delays in call setup, the 3PCR platform should quickly respond to any SIP INVITE that it receives. It is not expected that the 3PCR platform will send any 18x responses, but instead, that it immediately responds with a *200 OK* to an INVITE request. In the *200 OK* response, the codecs in the SDP should be a subset of the codecs in the initial offer, and they should contain all of the codecs that are supported by the 3PCR platform from the offered set.

4.5 Metadata Updates

The Application Server can send updates to the 3PCR platform if information in the call recording changes. The updated metadata can be received in an UPDATE or INVITE message.

Typically, metadata updates occur if the subscriber with the Call Recording service is the controller of a network-based conference call, with a Cisco BroadWorks-provided conference bridge.

Another scenario that results in metadata updates is during consultative transfer, since it behaves the same way as it does during a conference call.

Metadata updates can also occur:

- When the user with the Call Recording service calls a Call Center or Route Point, and the call is sent to an Agent to be answered.
- When the user with the Call Recording service is parked and then retrieved.

> **NOTE**: If the retrieving party is the *same party* that parked the call, the updated metadata will reflect this party as two separate participant entries in the metadata, with one entry as an active participant with <send> data, and the second entry without <send> data. This conveys that the retrieving party was initially in the recorded conversation prior to the call being parked, and then returned back in the recorded conversation after the call was retrieved.

4.5.1 Release 20.0 Changes

4.5.1.1 Start, Stop, Pause, and Resume

There are no changes to the metadata with the introduction of this feature. Since the call recordings in both of the new recording modes, *Always with Pause/Resume* and *On Demand with User initiated Start*, are kept, they are both reported to the 3PCR platform with the recording type of *On.*

4.5.1.2 Video Support

The Cisco BroadWorks Telephony Application Server can send updates to the 3PCR platform if information in the call recording changes. The updated metadata can be received in an UPDATE or INVITE message. All of the existing scenarios that send metadata updates as defined for audio-only calls are also applicable for calls that include video. Only those cases that also trigger metadata updates due to video are specified here. In general, any time there is a change in the SDP of the sessions that involve video, the 3PCR is updated with the changes.

Some typical cases of metadata updates that are triggered specifically for video calls are as follows:

- Subscriber with the Call Recording service is in an audio-only call with recording in progress and then adds video in the middle of the session. The 3PCR is re-invited to add video to the session. This applies to various call topologies such as two-party calls, three-way conference, and N-Way conferences.
  - If the video sessions fail to set up with the Media Server for any reason and the 3PCR sessions are set up with video prior to the failure, then 3PCR is re-invited to remove video from the session.
  - If the video session renegotiations fail between the clients for any reason and the 3PCR sessions are set up with video prior to the failure, then 3PCR is re-invited to remove video from the session.
- Subscriber with the Call Recording service is in a video call with recording in progress and then removes video in the middle of the session. The 3PCR is re-invited to remove video from the session.

4.5.1.3 Voice Mail Recording

There is a change to the metadata due to this feature. The recording type in the Cisco BroadWorks extension now allows the new “voicemail” type.

4.5.1.4 Visual Security Classification

This feature enhances the existing Call Recording service by adding a new Cisco BroadWorks-specific metadata element that is sent to the 3PCR platform for recordings of subscribers assigned with the Security Classification service.

The Cisco BroadWorks-specific extension metadata now contains the new element, *callClassification*, which is the highest security classification level reached for the call.

4.6 Loss of Real-Time Transport Protocol Stream

In the event that the RTP stream for a call recording is terminated, the 3PCR platform should consider this to be the end of the call and should immediately stop recording the call. This situation could occur when the Application Server fails over to a secondary server. The call that was in the process of being recorded has its media paths torn down and the call ends abruptly.

5 Call Recording Feature Details and Service Interactions

5.1 Recording Modes

The service can be configured to record all calls, or to selectively record calls triggered by a user's input, or to never record calls that a user makes or receives. The following five settings control the recording behavior: *Always*, *Always with Pause/Resume*, *On-Demand*, *On-Demand with User Initiated Start*, and *Never*.

5.1.1 Always Mode

If the user's recording mode is *Always*, then the Call Recording service automatically records all calls to the 3PCR platform without the user taking any action. For any calls that the user originates/receives/joins, the Call Recording service makes sure that the audio and video, if enabled, for these calls are recorded and saved to the 3PCR platform.

5.1.2 Always with Pause/Resume Support

The *Always with Pause/Resume Support* mode enables the pause/resume functionality. Calls that belong to users with this mode are always recorded automatically. However, they have the ability to pause and then resume the recording.

5.1.3 On-Demand Mode

If the user's recording mode is *On-Demand*, then the Call Recording service records the calls, but only the recordings of those calls that the user triggers with a FAC (*44) or with an Xsi command from a remote application are kept by the 3PCR platform.

When using the *44 FAC, the call that is kept is the one that was most recently put on hold.

When using the Xsi command, the call ID for the desired call to be kept is sent in the Xsi record request.

Once a call recording has been marked as kept, the pause/resume functionality becomes active. Recording can be paused or resumed.

5.1.4 On Demand with User Initiated Start

The *On Demand with User Initiated Start* mode added in Release 20.0 differs from all of the other modes in that the recording of the call is not started until the user starts recording the call. The call recording can be started using the *44 FAC or the Xtended Services Interface *record* event. The setup for the recording occurs as soon as the parties are successfully connected together. For example, if the user dials the *44 FAC to record the most recent locally held call, it is not until the user returns to the locally held call that the recording is set up for that call.

Once the call is being recorded, the pause/resume and stop functionality become available. The user can stop the recording with either the *45 FAC or the Xtended Services Interface *stopRecording* event. Once the recording is stopped, the connection to the 3PCR platform is disconnected. The user can start a new recording after the previous recording has stopped.

5.1.5 Never Mode

If the user's recording mode is *Never*, then no calls are recorded for the user. Although the service is active for the user, nothing is ever recorded, even if the user attempts to trigger the recording via the FAC or Xsi command.

It is possible that all users on a call may have the Call Recording service assigned and active. In these cases, a separate call recording is made for each of the users that have the Call Recording service active.

When the service is active for a user, both originating and terminating calls are recorded for the user. Call recording starts when the call is answered. If the user has the *On-Demand* option selected, the call is recorded but the recording is not kept by the 3PCR platform unless the user informs Cisco BroadWorks that they want to keep the recording. This can be done by dialing a FAC (*44) or through a client that implements the Cisco BroadWorks Xtended Services Interface extension, as defined in this document. Notification that the user wants to keep the recording must be sent to Cisco BroadWorks prior to the call ending.

The recording starts after the far-end party answers the call and media renegotiation takes place. A call is considered answered when the far-end party has sent a *200 OK* response to the INVITE message and has received the ACK. Once a call has been answered, a SIP re-INVITE is sent to both parties to redirect the media streams to the Media Server for streaming to the Call Recording Server. The call starts recording after the media renegotiation completes.

5.1.6 Recording Started Notification in Always, Always with Pause/Resume, and On Demand Modes

A new configuration option *Play Call Recording Start/Stop Announcement* enables the *Recording Started* and *Recording Ended* notifications in the *Always*, *Always with Pause/Resume*, and *On Demand* recording modes. The option is disabled by default.

In these recording modes, the call recording starts automatically at the beginning of the call. With the option enabled, when the user makes or receives a call, as soon as the call is connected and the call recording starts successfully, the *Recording Started* announcement is played to notify all parties that the call is being recorded.

5.1.7 Recording Started/Ended Notification in On Demand with User Initiated Start Mode

The new configuration *Play Call Recording Start/Stop Announcement* option enables the *Recording Started* and *Recording Ended* notifications in the *On Demand with User Initiated Start* recording mode. The option is disabled by default.

In this recording mode, the call recording does not start until the user initiates it with the FAC or the Xtended Services Interface command. With the option enabled, when the user initiates call recording, the *Recording Started* announcement is played to all parties on the call. When the user stops recording the call, just before the users are released from the media connections and the call recording stops, the *Recording Ended* announcement is played to all parties on the call to notify them that the call recording has ended.

A user uses the Xtended Services Interface command to start and stop call recording.

- Use Xtended Services Interface record command to start call recording. The *Recording Started* announcement is played to all parties on the call.
- Use Xtended Services Interface *stopRecording* command to stop call recording.

  The *Recording Ended* announcement is played to all parties on the call.

A user uses the *44/*45 FAC to start/stop call recording.

When the user dials the FAC in the middle of the call, all other parties on the call are put on hold. It is assumed that that the Music On Hold is not enabled; otherwise, the music that is on hold overrides the notification announcements.

- Dial *44 + [destination number] to start recording at the beginning of the call.
  - The Call Recording service is activated and the user hears the FAC success announcement.
  - Then the call terminates to the destination party.
  - After the destination party answers and the call recording starts, both parties hear the *Recording Started* announcement.
- Dial *44 to start recording in the middle of the call.
  - The Call Recording service originator first hears the FAC success announcement.
  - Then the Call Recording service originator resumes the held call and all parties on the call hear the *Recording Started* announcement.
- Dial *45 to stop recording.
  - The Call Recording service originator hears the FAC success announcement.
  - All other parties on the call hear the *Recording Ended* announcement.

5.1.8 Recording Ongoing Notification in All Modes

A new configuration option *Repeat Record Call Warning Tone* enables the *Recording Ongoing* notification in all recording modes. The option is disabled by default.

Another new configuration parameter, *Repeat Record Call Warning Tone Timer*, controls how often the end user receives a *Recording Ongoing* notification indicating that the call is being recorded. The parameter ranges from 10 through 1800 and it is set to "15" by default. It is used to set the repeating timer for the *Recording Ongoing* notification after the call recording starts.

For example, with the *Repeat Record Call Warning Tone* option enabled and the *Repeat Record Call Repeat Warning Tone Timer* parameter set to "15", 15 seconds after the call recording starts or after the *Recording Started* announcement (if applied) is played, the *Recording Ongoing* warning tone is played to all parties in the call to remind them that the call is still being recorded. The warning tone is repeated every 15 seconds until the call is released or until the user stops the call recording.

5.1.9 Calls Originated by Recording User

When the recording user originates a call, the call recording starts as soon as the remote party answers. Depending on when the call recording starts, any announcements or early media played prior to the call being answered are not recorded. This may include interactions with an Interactive Voice Response (IVR) if the system does not answer the call prior to the IVR interaction.

The recording does not start until the remote party answers the call; this is a restriction and it includes Click-To-Dial-type calls. In a Click-To-Dial scenario, the Application Server initiates a call to the originator and then redirects the originator to the destination the originator intended to call. The recording does not begin until the redirected call is answered.

In addition, if the call is rejected by the far end, any treatments played to the originating user are not recorded.

5.1.10 Calls Received by Recording User

An inbound call is a call that the user who has the Call Recording service is answering. The recording of these inbound calls starts after the user has answered the call. Any media provided prior to the user answering the inbound call is not recorded.

5.1.11 Codec Selection for Recording

The codec selected for the recording is dependent on the capabilities of the 3PCR platform and the end-user devices. The Cisco BroadWorks Media Server, which streams the RTP media to the 3PCR platform, can transcode the media, if necessary, to ensure that a recording is made. The Cisco BroadWorks Media Server should be configured to support the broadest range of audio codecs possible. The codecs supported by the Media Server are dependent on the network in which the system is deployed and are also dependent on any audio codec bandwidth limitations by the vendor and clients in use.

For information on configuring the codecs supported by the Cisco BroadWorks Media Server, see the *BroadWorks Media Server EMS Configuration Management Guide* [9].

5.1.12 DTMF Digits

A call recording can contain the dual-tone multi-frequency (DTMF) digit sequences input by the parties on the call, but only if the digits are contained in the RTP streams. However, if the 3PCR platform does not support *RFC 2833* [6], *RFC 4733* [7] or *RFC 4734* [8], then DTMF digits transported in the RTP stream using these mechanisms are not recorded. In addition, if the DTMF digits are transported in the body of a SIP INFO message, they are also not part of the call recording.

5.2 External Tracking IDs

The metadata that is sent to the 3PCR platform includes an external tracking ID. This ID is only unique for a given subscriber, and this ID correlates the calls for the user. This ID can change over the life of a call recording. For example, when a call is transferred, the external tracking Id is updated to reflect the new call.

5.3 Recording Behavior

When the call is recorded and for which users on the call it is recorded follows the existing behavior of audio-only call recordings.

The recording starts after the far-end party answers the call and media negotiation with both audio and video is completed. A call is considered answered when the far-end party has sent a *200 OK* response to the INVITE message and has received the ACK. Once a call is answered, a SIP re-INVITE is sent to both parties to redirect the media streams to the Media Server for streaming to the Call Recording Server. The call starts recording after all media renegotiation completes.

If the 3PCR is already recording video and if the end users on the call make any changes to the video SDP triggering a renegotiation between users, the 3PCR is also updated with the changes.

5.3.1 Calls Originated by Recording User

When the recording user originates a call, the call recording starts after the remote party answers. Any early media on the call is not recorded.

5.3.2 Calls Received by Recording User

An inbound call is a call that the user who has the Call Recording service answers. The recording of these inbound calls starts after the user answers the call. Any media provided prior to the user answering the inbound call is not recorded.

5.4 Redundancy and Load Balancing

5.4.1 Media Server Redundancy

Release 21.0 introduces a new feature to support Media Server Redundancy.

When the Call Recording service is notified of the connection failure of the Media Server, it attempts to reconnect to an alternate Media Server. If the call recording has not started yet as in the case of a user with *On-Demand User Initiated Start* that has not started recording, then no action is taken. In all other cases, the following actions are taken by the Call Recording service:

1) Call recording attempts to locate an alternate Media Server.

2) Once an alternate server is located, the Call Recording service reconnects the parties in the call to the Media Server.

3) The Call Recording platform server is updated with the SDP from the new Media Server.

If the recording was paused, it stays paused on the new Media Server until the recording is resumed via a FAC or Xtended Services Interface (Xsi) action. If the mode is *On-Demand* and the user has not sent the Xtended Services Interface record action or the *44 FAC, the restored call recording session is still in that state. If the user wants to keep the recording, then they must still enter the FAC or send the Xtended Services Interface record action.

If the new Media Server does not support video or does not have capacity to accept an additional video call, then when call recording resumes on the new Media Server, the recording changes to an audio-only recording. The Call Recording platform server is informed of this by the change in the metadata and SDP sent to the platform server.

If it is not possible to connect to an alternate Media Server, then the behavior of the Call Recording service depends on the settings of the *continueCallAfterRecordingFailure* and the *continueCallAfterVideoRecordingFailure* system parameters when the mode is *Always* or *Always with Pause/Resume*. For the other modes, whether the above parameters are followed are dependent on *useContinueCallAfterRecordingFailureForOnDemandMode* and *useContinueCallAfterRecordingFailureForOnDemandUserInitiatedStartMode* parameters, if they are set to “false” then the calling and called parties are reconnected and call recording stops; otherwise, they also follow the continue call behavior specified in the two system parameters.

How much of the call recording is lost when the Media Server loses connection depends on what caused the failure of the Media Server and how long the detection of the loss takes. For this reason, it is recommended that the *crAuditIntervalMilliseconds* and *crAuditTimeoutMilliseconds* be set to relatively small values to reduce this period of lost voice path.

While the Application Server is in the process of reestablishing the signaling and media paths through the alternate Media Server, any recording actions invoked by the user (via Xtended Services Interface or FAC) are processed seamlessly. The recording action is immediately acknowledged; however, the action (for example, pause and resume) itself takes effect once the connections to the alternate Media Server have been established.

For more information, see the *Media Server Redundancy for Call Recording and N-Way Conferencing Feature Description* [24].

5.4.2 Load Balancing Using 302 Redirection Support

The Call Recording platform can point to a redirection server for load balancing between multiple Call Recording servers. Release 21.0 and onwards supports a redirection event in the Call Recording Manager. When a 302 response is received, the Application Server pulls a list of contacts from the 302 response and then attempts to connect to each server in turn based on the q-value and order of the contacts.

If all the servers returned in the 302 redirection attempts fail, then the Application Server marks the Call Recording Platform as failed.

If the Call Recording Platform redirects to another server that then redirects to yet another, then the Application Server allows a total of three redirections. Once the platform has been redirected three times and fails, then the Application Server marks the platform as failed and generates a log.

5.5 Video Call Support

The Cisco BroadWorks Application Server and Media Server provide support for Call Recording for Video.

5.5.1 Video Fast Update Support

The Cisco BroadWorks Telephony Application Server proxies a SIP INFO request that includes a *picture_fast_update* primitive as defined in *RFC 5168* [20] to the clients, when renegotiations take place between users and call recording setup is completed. This forces the clients to send a video refresh, thus minimizing the period where no image is displayed at the far end. The following is an example of a SIP INFO that includes a *picture_fast_update* primitive.

```
INFO sip:5146992503@192.168.8.97:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.200.205:5060;branch=z9hG4bK-BroadWorks-MS-
329699111
From: <sip:5678@mtlasdev92.net>;tag=89314878
To: <sip:5146992503@mtlasdev92.net>;tag=499861f78d7f218c
Call-ID: f641e626079e46d8@192.168.8.97
CSeq: 435582615 INFO
Content-Type: application/media_control+xml
Max-Forwards: 70
Content-Length:   169

<?xml version="1.0" encoding="utf-8"
?><media_control><vc_primitive><to_encoder><picture_fast_update></picture
_fast_update></to_encoder></vc_primitive></media_control>
```

5.5.2 Video Codec Negotiation

The list of video codecs used by the Media Server for Control Channel Framework (CFW) services is now configurable independently from other IVR services. Prior to this feature, the Media Server shared the list of supported video codecs with other IVR services. A new command line interface (CLI) level allows adding and removing video codecs that can be negotiated with Control Channel Framework services.

The following video codecs are configurable:

- H264

- H263-1998
- H263-2000

Note that video conferences on the Media Server only support H264. For more information on the Media Server video conference support, see the *Video Conferencing Support Feature Description* [21].

5.5.2.1 Video Codec Selection for Recording

The codec selected for the recording is dependent on the capabilities of the 3PCR platform and end-user devices. It also depends on the codecs that the Media Server supports. For dual mode, the Media Server supports H.264 and H.263 codecs. In single mode, it supports only the H.264 codec. For more information on the codecs supported by the Cisco BroadWorks Media Server and how to configure them, see the *BroadWorks Media Server EMS Configuration Management Guide* [9].

The 3PCR must adhere to the following rules when handling video SDP to successfully record video calls:

- In dual mode, it must support the H.264 and H.263 video codecs. In single mode, it must support H.264 codec.
- It must support the same level of resolutions that are offered to it without downgrading them in the answer.
- If the Media Server is NOT Release 20.0 with asymmetrical payload support: It must support symmetrical payload numbers, which means that it must respond with the same payload type number in the answer as in the offer.

The Cisco BroadWorks Media Server, which streams the RTP media to the 3PCR platform, does not transcode the video media. It supports transcoding for audio only. This necessitates that the clients do end-to-end media negotiation first and the call recording associated renegotiations are all initiated after answer of the call between users. Note that to successfully record video, the offer video SDP from the end user must have at least one video codec format that is supported by the Cisco BroadWorks Media Server.

The following sections describe each of the 3PCR platform's modes and assume that the call is a two-party call. User A is the originator and User B is the terminator. Note that only video SDP is described.

5.5.2.1.1 Video Codec Selection in Dual Mode

SDP to Call Recording Platform

The SDP sent to the Call Recording platform is based on the SDP from the end users and the codecs supported on the Media Server.

To compute this list of codecs, the following filtering is applied:

- The offer from the Media Server is used as the base SDP, which is filtered to include only those codecs that were negotiated between User A and User B. From this set, the codec with the highest preference is selected.
- Any *fmtp* attributes from the SDP of User A associated with the selected codec are also included.
- Any generic video attributes from the SDP of User A are also included. This is the resulting offer with the label attribute 3 in the SDP sent to 3PCR.

The previous three steps are repeated for the SDP of User B and the resulting offer with the label attribute 4 in the SDP is sent to 3PCR.

This is the video SDP that represents User A’s and User B’s offer to the 3PCR platform.

SDP to User A and User B

When reconnecting users for call recording, Users A and B are re-invited by the Cisco BroadWorks Telephony Application Server so that the SDP is streamed through the Media Server. During the renegotiation, the offer/answer SDP from the Media Server and the original call between Users A and B is used as the base.

The following rules are applied to arrive at the SDP used in the renegotiation.

Offer to B:

The offer SDP sent to User B is the SDP from the Media Server filtered through the negotiated SDP from the original call setup with the attributes from SDP A copied as explained above.

Answer to A:

The answer SDP sent to User A is the answer from the Media Server with the attributes from the answered SDP User B copied, as explained above.

An example of a video call recording being set up between Users A and B, which shows the rules described above being applied to the SDP is described in section *10.3.2 Basic Video Call Recording in Dual Mode*.

5.5.2.1.2 Video Codec Selection in Single Mode

In single mode, the SDP sent to the 3PCR is the offer SDP from the Media Server. Unlike dual mode, the SDP is not manipulated since the Media Server mixes the output from the users and regenerates the video stream sent to the 3PCR. If multiple codecs are offered by the Media Server, the codecs are reordered to match as close as possible to the original negotiated codec list between Users A and B.

The offer from the Media Server sent to User B is reordered in a manner that is similar to that shown above. Note that for this release, the Media Server only supports H.264 codec for single mode and, as a result, there is no reordering involved.

For the call flow for single mode call recording, see section *10.3.3 Messaging to Show SDP Negotiation*.

Recording Platform Provisioning for Video

Call Recording platform provisioning includes new parameters to configure the support for video recording.

The *supportVideoRecording* parameter is used to enable or disable the video recording functionality. This parameter is defined at the Call Recording platform level and is disabled by default. This parameter needs to be enabled to process video calls for recording. When this parameter is disabled, the video media is streamed between end users, but only the audio stream is recorded.

The *continueCallAfterVideoRecordingFailure* parameter is used to define the behavior when video recording is enabled and video recording fails for any reason. This parameter is defined at the Call Recording service level and is enabled by default. When disabled, if the video recording fails for any reason, the call between the users is also terminated. If enabled, the call between the users is allowed to continue without video being recorded. For more information, see section *5.5.3 Video Recording Failure Handling*.

5.5.3 Video Recording Failure Handling

The Cisco BroadWorks Telephony Application Server may be unable to successfully record video for the following reasons:

- Video SDP negotiations with the 3PCR platform or the Media Server may fail.
- The Media Server may fail to join the video streams.
- The renegotiation of video SDP with the end users may fail.

The behavior of the Cisco BroadWorks Telephony Application Server when a failure to record occurs is dictated by the system-level parameters *continueCallAfterVideoRecordingFailure* for video and *continueCallAfterRecordingFailure* for audio. Note that these parameters are valid for all modes but whether the two on demand modes follow them depends on the following two system parameters: *useContinueCallAfterRecordingFailureForOnDemandMode* and *useContinueCallAfterRecordingFailureForOnDemandUserInitiatedStartMode.*

- *continueCallAfterRecordingFailure* = “true”

  When this option is enabled, the *continueCallAfterVideoRecordingFailure* is ignored. Any call recording failure, be it audio or video, results in the call continuing between end users.

  - Failure to successfully negotiate video with the 3PCR platform or the Media Server.

    Audio streams are recorded. The video is streamed between users but is not recorded.

  - Failure to successfully renegotiate video with the end users.

    The 3PCR platform is re-invited to record audio streams only.

  - Failure to successfully negotiate audio and video with the 3PCR platform or the Media Server.

    The call is continued between users and media and is streamed directly between them.

  For call flows, see section 10 Call Flow Details.

- *continueCallAfterVideoRecordingFailure* = “true”; *continueCallAfterRecordingFailure* = “false”

  - If video negotiation fails at any point in the setup sequence, the audio streams are recorded and the call continues.
  - If audio negotiation fails at the 3PCR, Media Server, or users, the call is terminated.
  - If the call is rejected entirely by either the Media Server or 3PCR, the call between users is terminated as well.

- *continueCallAfterVideoRecordingFailure* = “false”; *continueCallAfterRecordingFailure* = “false”

  If audio, video, or the entire call is rejected, the call is terminated, and resources are released.

5.6 Video Add-On Service

The Video Add-On service enables the use of video media in conjunction with regular audio media. If a user's primary device does not support video, this service can be used to configure a video-capable device to deliver the video portion of their call. When the user receives a multimedia call, Cisco BroadWorks divides the call, directing the audio portions of the call to the primary device and the video portions of the call to the video add-on device.

If call recording is enabled on such a call with a video add-on device, the video portion of the call is also recorded in addition to the audio.

5.7 DTMF Transmission Service

It is important to note that this service is now supported for a user with the Call Recording service enabled.

5.8 In-band versus Out-of-band DTMF Transmission

If the users being recorded support using different methods to transmit DTMF digits then when the Call Recording service starts up, the DTMF transmission mode is the same on both call halves based on the original end-to-end call negotiation. The capabilities of this call setup determine whether out-of-band DTMF is supported when the call recording starts.

5.9 Conference Call

5.9.1 Three-Way Conference

Three-way conference calls can pose a problem since there are two common ways the calls are bridged. Many devices are capable of setting up three-way conference calls without the need for a network conference bridge. In these cases, if the user's device bridges the calls together, they appear as two separate call recordings on the Call Recording Server. There is no indication that the two calls are part of a conference call. The second case is when Cisco BroadWorks provides the conference bridge. In this case, all of the participants of the conference are listed in the XML extension data.

5.9.2 N-Way Conference

When a user with the Call Recording service is the controller of an N-Way conference call, all of the calls in the conference generate their own recording. The recording starts when the user places or receives a call from each party participating in the conference call. When the parties are transferred into the N-Way conference, the XML extension data for each call contains the list of all the other participants in the conference. As each call is transferred into the conference, the call being recorded changes from being the conversation between the conference initiator and the party, to being a recording of the conference call.

To illustrate the above scenario, assume the following:

- User A has the Call Recording service and sets up a conference call between Users B, C, and D.
- After two minutes, User E is added.
- After two more minutes, User C drops out of the conference.

The following recording results:

1) Call between User A and User B – Contains the conversation between User A and User B prior to the conference, and then after the conference starts, it contains the conversation on the bridge.
2) Call between User A and User C – Contains the conversation between User A and User C prior to the conference, and then after the conference starts, it contains the conversation on the bridge up until the time that User C leaves the conference.
3) Call between User A and User D – Contains the conversation between User A and User D prior to the conference, and then after the conference starts, it contains the conversation on the bridge.
4) Call between User A and User E – Contains the conversation between User A and User E prior to User E being brought into the conference, and then it contains the conversation on the conference bridge starting two minutes later than the other parties.

If User A's recording mode is *On-Demand* and if User A is the controller of the N-Way conference, the *44 FAC code can be dialed to trigger the N-Way conference call to be saved, as long as the N-Way conference was the most recently held call. All call legs that belong to User A's N-Way conference will be saved.

If User A is *not* the controller of the N-Way conference, the *44 FAC code will only record his leg of the N-Way conference. It will not record the other parties' legs in the N-Way conference.

Since the Xsi command is only intended to record one call at a time with a given call ID, it will only record the specified call in the N-Way conference that matches the call ID that is provided in the Xsi request.

5.9.3 On Demand with User Initiated Start

The *On Demand with User Initiated Start* mode introduces the ability to start the call recording at any point in the call. This changes how individual calls in a conference call are recorded by the conference controller. The conference controller is the party who initiates the conference.

The descriptions in this section refer to a Cisco BroadWorks user who is initiating the N-Way/Three-Way Conference. The conference controller in each of the following descriptions is User A. If the user is a participant in the conference but not the controller, then the recording follows the same rules as for a two-party call.

5.9.3.1 Start Call Recordings

The following subsections describe the effect of starting the call recordings at various points in the conference call.

5.9.3.1.1 Conference Active

When a conference call has already been established and the user initiates call recording, then there is only a single recording session started. It records the conversation between the conference initiator and the other parties in the conference. This recording is started as a recording between User A and the participants of the conference. The call that is recorded is identified by the Xtended Services Interface event.

Since this call recording is associated with only one of the participants on the conference, if that participant leaves the conference, the recording stops. This differs from the other call recording modes, since in the other modes the recordings start recording on all of the participants when the initial calls are set up. When one participant leaves, the other recordings continue, since there are multiple recordings, one for each participant of the conference.

There is also a change in behavior when a party is added to a conference that is being recorded. If User A is already in a conference with Users B and C, and then decides to add User D to the conference, User A calls User D. The call between User A and User D is not recorded, since User A did not initiate the call recording. However, when User A adds User D to the conference, User D now becomes part of the recorded calls. It is up to User A to inform User D that they are being recorded since no announcement or other indication is played to User D indicating that their call is now being recorded.

When the user initiates the call recording of the conference call using the *44 FAC, the recording is started if the conference call is the most recent locally held call. In this case, all of the participant calls are selected by the Telephony Application Server and recording is started on each of the calls. In this case, if one participant releases from the conference, that recording stops but the recording of the other participants continues.

5.9.3.1.2 User A to User B Call Recording and User C Conferenced In

A second scenario that the new *On Demand with User Initiated Start* mode makes possible is to be recording one call while adding a second call that is not being recorded, and then initiating a conference. Prior to this mode being available, all of the call legs would be recorded prior to the conference being initiated.

In this case, User A initiates a call to User B and starts recording the call. At some point in the conversation, it is decided to bring User C onto the call. Therefore, User A calls User C and then initiates a network conference call. Since User A is recording the call between User A and User B, the recording should continue on the conference call. Therefore, once the conference is established, the call recording continues; however, it is now recording User A’s conversation with Users B and C. The metadata for the recording is updated to include User C.

If User B leaves the call, then call recording stops since User A was only recording the call between themselves and User B.

5.9.3.1.3 Conference Active and Recording and User D Added to Conference

This scenario involves adding a new participant to an N-Way conference when the conference is already established and is being recorded. In this case, User A decides to add User D to the conference. Therefore, User A puts the conference call on hold. The call recording does not pause when User A puts the call on hold. It continues to record the conference bridge capturing any ongoing conversation between User B and User C. User A initiates a call to User D. User A then bridges User D into the conference. The call recording continues and users A, B, C, and D are now all in the conference. The metadata for any call recordings is updated to include User D.

In this case, the number of call recordings depends on how many there were prior to User D being added. Adding User D to the conference does not change the number of call recordings.

The call recording continues until all users being explicitly recorded are released from the call or User A stops the recording.

5.9.3.1.4 Conference Active, User A to User D Call Started and Recording, Then Added to Conference

This scenario is similar to the previous scenario except in this case User D is added to the conference after starting to record the call between User A and User D. Prior to user D being added, there is a conference between Users A, B, and C with User A not recording any of the participants. User A decides to add User D to the call. User A puts the conference call on hold and then initiates a new call to User D. After User D answers, User A starts the Call Recording service. After the call recording starts, User A bridges the call into the existing conference. This causes the call recording to start recording the conference call on behalf of User A. The metadata for the call recording is updated to include User B and User C as participants.

If the conference was already being recorded, adding User D would add one additional recording to the conference call. All of the recordings would be on behalf of User A, who in this case is the conference controller. The other users can be recording the call as well; however, the recordings are tracked per user and do not affect each other.

The call recording continues until all parties being recorded by User A release the call or User A stops the recording.

5.9.3.2 Stop Conference Call Recording

When the conference controller enters a *45 FAC for a conference call, all call recordings active on the conference are stopped. There is no way for the FAC to identify the particular call for which it was started.

However, with the Xtended Services Interface *stopRecording* request, the call ID of the call that is being recorded is specified so that the individual call recording can be stopped. If there are multiple call recordings active on a conference call (for the controller) and if the Xtended Services Interface action is used to stop the recording, only the single recording specified in the action is stopped while the other call recordings remain active.

5.9.4 Pause/Resume

The behavior of the pause/resume functionality differs based on the method used to pause or resume the recordings. This section only applies to the controller of the conference call.

5.9.5 Video Conference

Call recording for three-way and N-Way conference calls behaves similar to calls with only audio except that video is also recorded if the video codec negotiations between the end users, the Media Server, and the 3PCR platform are successful. If the video negotiations fail, then only the audio portion may be recorded depending on the configuration settings.

If the user being recorded downgrades from an audio/video stream to an audio-only stream, the 3PCR is first put on hold while reconnections are made with the Media Server and then it is re-invited with an updated SDP to indicate the removal of the video streams.

If the conference being recorded downgrades from audio/video stream to an audio-only stream, the 3PCR is similarly updated with the changes.

To illustrate the previous scenario, assume the following:

- User A has the Call Recording service and sets up an audio/video conference call between Users B, C, and D. Users A and C are sending audio and video. Users B and D are sending only audio.
- After two minutes, User A drops the video.

- After two more minutes, User C drops the call.
- After one minute, Users A and B add the video.

Note that putting the 3PCR on hold while reconnecting to update the SDP may cause a pause in the recording; however, this is not included here as it is not expected to be a significant pause.

The following are the results:

- Call between User A and User B – Contains the audio between User A and User B prior to the conference, and then after the conference starts, it contains the audio with video on the bridge. After two minutes, the recording has audio from User A and audio from the bridge. After two more minutes, the recording only contains audio. Then after one minute, the recording has audio from User A and audio/video from the bridge.
- Call between User A and User C – Contains the audio with video between User A and User C prior to the conference, and then after the conference starts, it contains the audio with video on the bridge. After two minutes, the recording has audio from User A and audio from the bridge. After two more minutes, the recording stops.
- Call between User A and User D – Contains the audio between User A and User D prior to the conference, and then after the conference starts, it contains the audio with video on the bridge. After two minutes, the recording has audio from User A and audio from the bridge. After two more minutes, the recording only contains audio. Then after one minute, the recording has audio from User A and audio/video from the bridge.

A conference can start out with all participants using audio-only and then upgrade to video if any of the conference participants add video. In this situation, if the call recording is active for the users, their sessions with the 3PCR are renegotiated to add video so that video can also be recorded.

There are no changes to how call recording behaves for adding or removing participants from the audio-only conferences.

Call recording manages Meet-Me conferences similar to how it manages a two-party call, and as a result, the video recording behavior follows that of a two-party video call.

For a call flow describing a three-way conference, see section *10.3.7 Video Conferencing*. For video SDP negotiation failure handling, see section *10.3.3 Messaging to Show SDP Negotiation*.

5.9.6 End-User Notification

5.9.6.1 Three-Way Conference

A user uses the device with the local conference capability to set a local conference. In this situation, the local conference appears to have two separate calls. The end-user notification behaves as a normal two-party call.

For a user establishes a network conference, the end-user notification behavior is similar to the N-Way conference.

5.9.6.2 N-Way Conference

When a user with the Call Recording service in the *Always* mode sets up an N-Way conference call, all calls in the conference generate their own recording. The recording starts when the user places or receives a call from each party participating in the conference call. When the parties are transferred into the N-Way conference, the XML extension data for each call contains the list of all the other participants in the conference. As each call is transferred into the conference, the call being recorded changes from being the conversation between the conference initiator and the party, to being a recording of the conference call.

In the following subsections, the end-user notification behaviors are illustrated based on the call recording modes.

5.9.6.3 Always, Always with Pause/Resume, and On Demand Modes

Assume that User A has the Call Recording service unless otherwise specified.

User A initiates a conference

1) User A talks to User B and the call is being recorded.
2) User A invites User C and establishes a conference with User B and User C.

   The second call recording starts between User A and User C when User C answers. The *Recording Started* announcement is played to User C when User C answers.
3) User A invites User D to the conference.

   The third call recording starts between User A and User D when User D answers. The *Recording Started* announcement is played to User D.

The *Repeat Record Call Warning Tone* option is enabled in the above conference. After the conference is sent up, the recording warning is played to the conference in the provisioned frequency. Only one copy of warning tone is played even though there are multiple call recordings ongoing in the conference.

> **NOTE**: If the call recording party is not the conference controller, the recording call is treated as a regular two-party call in which one is the call recording party and the other is the conference. When the call recording party leaves the conference, there is no Recording Ended announcement played to remaining conference parties.

5.9.6.4 On Demand with User Initiated Start Mode

User A initiates a conference and then initiates the call recording with the *44 FAC

When the user initiates the call recording of the conference call using the *44 FAC, the recording is started if the conference call is the most recently held call. In this case, all participant calls are selected by the Telephony Application Server and recording is started on each call.

1) User A is in the conference with User B, User C, and User D.
2) User A puts the conference on hold and dials *44 to start the call recording.

   Three call recordings start, one between User A and User B, one between User A and User C and one between User A and User D. The *Recording Started* announcement is played to the conference bridge.
3) User D leaves the conference. One recording stops but two recordings continue.
4) User C leaves the conference. Another recording stops but one recording continues.

5) User A is now talking to User B and the recording continues.

User A initiates a conference and then initiates the call recording with the Xtended Services Interface command

When a conference is active and a user uses the Xtended Services Interface command to start the call recording, only the call between User A and one other conference participant is recorded.

1) User A set up the conference with User B, User C, and User D.
2) User A uses the Xtended Services Interface command to initiate the call recording on the conversation between User A and User D. The *Recording Started* announcement is played to all parties on the conference bridge.
3) User D hangs up and leaves the conference. The call recording stops and there is no call recording in the conference. The *Recording Ended* announcement is played to all remaining parties on the conference bridge.

User A initiates a conference and starts the call recording on the conference invitation call

1) User A is in the conference with User B and User C. The call is not recorded.
2) User A makes a call to invite User D to the conference. User D answers, but before User A brings User D to the conference, User A initiates the call recording. The *Recording Started* announcement is played to User A and User D.
3) User A brings User D to the conference with User B and User C. The recording continues on the conference call. The *Recording Started* announcement is replayed to User B and User C.
4) User C leaves the conference. The recording continues.
5) User D leaves the conference. The recording stops and the *Recording Ended* announcement is played to the remaining parties.

User A initiates the call recording first and then initiates a conference

1) User A is on a call with User B.
2) User A initiates the call recording. The recording between User A and User B starts and the *Recording Started* announcement is played to User A and User B.
3) User A invites User C and establishes a conference. The recording continues on the conference call. The *Recording Started* announcement is replayed to User B only when the conference is set up.
4) User A brings User D into the conference. The *Recording Started* announcement is replayed to User D only when User D joins the conference.
5) User D leaves the conference. The recording continues.
6) User B hangs up. The recording stops and the *Recording Ended* announcement is played to the remaining parties.

5.10 Call Park

The Call Park feature allows a user *to park a call* and then retrieve the call at a later time. The interaction of the Call Recording feature with Call Park can vary depending on whether the user is parking/retrieving the call or whether the user is the party that is being parked.

The interaction of the Call Recording service with Call Park for video calls is the same as that of audio-only calls except that video is also recorded.

5.10.1 Parked Party

If the party being parked records the call, there is a single recording as a result. The recording includes the time the user was parked, as well as any silence or music during the time the user is parked depending on the configuration of the Call Park service.

5.10.2 Parking Party

If the party parking the call is recording the call, then the call recording ends when the call is parked.

5.10.3 Retrieving Party

If the party retrieving the parked call has Call Recording, then the retrieval starts a new call recording, even if the retrieving party is the same user who parked the call.

5.11 Call Transfer

5.11.1 Blind Transfer

In a blind transfer, a user is transferred to another party, and the user performing the transfer does so without talking to the party receiving the call. Assuming each party in the blind transfer scenario has the Call Recording service, the service behaves as follows for each of the parties:

- User being transferred – The call recording with the original user ends and a new recording starts when the target of the transfer answers the call.
- User transferring the call – The call recording stops when the user hangs up.
- Target of the transfer – The call recording starts after the user answers the transferred call.

5.11.1.1 Video Support

The interaction of the Call Recording service with Blind Transfer for video calls is the same as that of audio-only calls except that video is also recorded.

5.11.1.2 End-User Notification Support

Assume that only User A has the Call Recording service and this user is in the *Always* mode unless otherwise specified.

User A as a transferring party

When the transfer is complete, the call recording stops and the *Recording Ended* announcement is played to the transferred party.

NOTE: The mechanism used to play the *Recording Ended* announcement is to delay the event which triggers the call recording resource to be released. However, in the above scenario, if User A has the Call Recording service in the *On Demand with User Initiated Start* recording mode and only one leg of two transfer calls is being recorded (due to the unbalance conditions) the delay Call Recording (CR) resource mechanism cannot be performed. Therefore, if the transferring party has the Call Recording service in the *On Demand with User Initiated Start* recording mode, when the transfer is complete, the *Recording Ended* announcement is not played to the transferred party and transfer target that are currently recorded.

User A as a transferred party

When the transfer is complete, the call recording continues and no announcement is played.

When the transfer target answers, the *Recording Started* announcement is played to the transfer target only and the announcement is not repeated to the transferred party.

User A as a transfer target

When User A answers, a new call recording starts. The *Recording Started* announcement is played to both parties.

5.11.2 Consultative Transfer

In a consultative transfer, the user transferring the call talks to the target of the transfer prior to completing the transfer. Depending on the client, when the target answers the call, the initiator can either complete the transfer or initiate a conference. If a conference is initiated, the Recording service follows the three-way conference behavior in section *5.9.1 Three-Way Conference*. If the initiator completes the transfer instead, then call recording behaves as follows (assuming each party in this scenario has the Call Recording service):

- User being transferred – The call recording continues with the far end of the call now connected to the target of the transfer.
- User transferring the call – The call recording stops when the transfer is completed.
- Target of the transfer – The call recording starts after the user answers the call from the user transferring the call. When the transfer completes, the recording continues with the party that was transferred.

5.11.2.1 Video Support

The interaction of the Call Recording service with consultative transfer for video calls is the same as that of audio-only calls except that video is also recorded.

5.11.2.2 End-User Notification Support

In a consultative transfer, the user transferring the call talks to the transfer target prior to completing the transfer.

User A as a transferring party

When the transfer target answers, a new call recording starts between User A and the transfer target, and the *Recording Started* announcement is played to both parties.

When User A releases the call to complete the transfer, the recording between User A and the transfer target and the recording between User A and the transferred party both stop. The *Recording Ended* announcement is played to both the transferred party and the transfer target.

If User A has the Call Recording service in the *On Demand with User Initiated Start* mode, the consultation leg of call is not recorded unless User A initiates recording.

> **NOTE**: Due to the same reason as with the blind transfer, if the transferring party has the Call Recording service in the *On Demand with User Initiated Start* recording mode, when the transfer is completed, the *Recording Ended* announcement is not played to the transferred party and the transfer target that are currently being recorded.

User A as a transferred party

Call recording continues after the transfer is completed. The *Recording Started* announcement is played to the transfer target only and the announcement is not repeated to the transferred party.

User A as a transfer target

The call recording starts when User A answers. The *Recording Started* announcement is played to User A and the transferring party.

The call recording continues when the transferring party releases the call to complete the transfer. The *Recording Started* announcement is played to the transferred party only and the announcement is not repeated to the transfer target.

5.12 Virtual Subscriber Services

The Call Recording service is also available to some virtual users, namely Call Centers, Route Points, and Auto Attendants. There is no concept of an *On-Demand* recording for a virtual user. The only way to activate the service for virtual users is to select the *Always* mode.

5.12.1 Auto Attendant

The Auto Attendant service provides an IVR interface for inbound calls. From the Auto Attendant, the call can be routed to many other entities ranging from a voice mail system, a call center, or a subscriber.

The Call Recording service records the interactions of a caller with the Auto Attendant. The recording starts shortly after the Auto Attendant answers the call. If the Auto Attendant is video-capable, only the audio portion of the call is recorded. The call recording contains the audio played to the caller and any responses from the caller. The only exceptions are DTMF digits. The digits entered by the client are recorded based on the capabilities of the 3PCR platform. For more information on DTMF digit recording, see section *5.1.12 DTMF Digits*.

The recording of the interaction with the Auto Attendant stops when the Auto Attendant transfers the call to the final destination. Once the call is transferred, a new call recording may start, but it is dependent on the subscriber, device, or service to which the call is transferred.

5.12.1.1 Video Support

If the Auto Attendant is video-capable, then the video and audio portions of the call are recorded.

5.12.1.2 End-User Notification

There is no announcement played for the call recording start/stop on behalf of Auto Attendant because the End-User Notification of Recording feature is not applied to the virtual user.

The end-user notification behavior on the Auto Attendant call is based on the call originator or the call destination.

Call originator with the Call Recording service in the Always mode

User A makes a call to the Auto Attendant. Once User A is connected to the Attn IVR, the first call recording on behalf User A starts. The *Recording Started* announcement is played to User A and overlapped with the Attn announcement.

After User A enters the destination User B's number, the call is transferred to User B. The first call recording on behalf of User A stops. The second call recording starts when User B answers. The *Recording Started* announcement is played to both User A and User B.

Call originator with the Call Recording service in the On Demand with User-Initiated mode

User A dials "*44" + [Attn number] to make a call to the Auto Attendant. Once User A is connected to the Attn IVR, the first call recording on behalf User A starts. The *Recording Started* announcement is played to User A and overlapped with the Attn announcement.

After User A enters the destination User B's number, the call is transferred to User B. The first call recording on behalf of User A stops. The *Recording Ended* announcement is played to User A. User B answers and talks to User A, but the call is not recorded.

Call destination with the Call Recording service in the Always mode

The call recording on behalf of the call destination party starts when the destination party answers. The Recording Started announcement is played to both originator and destination

5.12.1.3 Voice Mail Recording

An Auto Attendant that has both the Call Recording and Voice Messaging User services has the ability to record voice message deposits. If the administrator has *Record Voice Messaging* enabled, then the voice message deposits are recorded.

5.12.2 Call Center

Calls that are routed to a Cisco BroadWorks Call Center can be recorded. When a call is received by a Call Center, the policies of the Call Center determine how the call is handled. Typically, Call Center calls are placed in a queue and then offered to an agent upon being received by a Call Center. In general, the Call Center calls are recorded shortly after the Call Center answers the call and until the call is routed to an agent.

For more information on the Call Center service, see *Cisco BroadWorks Call Center Solution Guide* [10].

5.12.2.1 Video Support

The Call Center can also be configured to play video announcements and treatments for all the same scenarios in which audio is configured. If video is streamed in any of these scenarios, the video is also recorded along with the audio.

5.12.2.2 End-User Notification

There is no announcement played for the call recording start/stop on behalf of Call Center because the End-User Notification of Recording feature is not applied to the virtual user.

The end-user notification behavior on the Call Center call is based on the call originator or the destination agent.

When a Call Center call is presented to an agent or is routed to a destination party by the Call Center policies, a new call recording starts if the agent or the destination answers the call or after the whisper announcement is played, if provisioned. The *Recording Started* announcement is played to the parties of the call.

When a supervisor takes over a call from an agent, a call recording starts from the moment the supervisor takes over the call. The *Recording Started* announcement is played to all parties.

5.12.2.3 Voice Mail Recording

A Call Center that has both the Call Recording and Voice Messaging User services has the ability to record voice message deposits. If the administrator has *Record Voice Messaging* enabled, then the voice message deposits are recorded.

5.12.2.4 Busy Treatment

There are multiple policies that result in a call being routed to a busy treatment in a call center. Before busy treatment is applied, it can be configured to optionally play an announcement and then proceed to an action. If an announcement is configured, it is captured as part of the call recording. After the optional announcement, the following actions are supported:

- Busy Treatment – If the busy treatment is configured to play an announcement prior to routing to busy treatment, the announcement is recorded. Once the call routes to the busy treatment, the call recording stops since the busy treatment is not played by the Call Center application.
- Voice Mail – The call is routed to the Voice Mail service and the call recording stops.
- Call Forwarding Busy (CFB) – If there was an announcement prior to routing to CFB, then the announcement is captured in the recording. When the call is routed to the Call Forwarding Busy destination, then the recording stops.

5.12.2.5 Transfer

Multiple policies also support transferring the call as part of the Call Center policy. Similar to the busy treatment, the transfer can be configured to play an optional announcement prior to transferring the call. In this case, the announcement is captured in the call recording and then the recording is stopped since the call is transferred to the configured destination.

5.12.2.6 Pre-queue Policies

When a call first enters a Call Center, there are several policies that can affect the call recording. The following list covers the behavior between the Call Recording service and the most common policies that run prior to the call being queued:

- Call Forwarding Always (CFA) – No recording is made because the call is immediately sent to the CFA destination.

- Holiday Service policy and Night Service policy – Both of these policies can be configured to support busy treatment or transfer, as described in sections *5.12.2.4 Busy Treatment*.
- Forced Forwarding – If Forced Forwarding is configured to play an announcement prior to transferring the call, then a recording of the announcement and the caller is made if the session with the 3PCR server is established prior to the announcement finishing.

5.12.2.7 Queued

This section describes the behavior of the calls as they enter the queue and how the various queue policies affect the call.

In general, the call recording starts when the call enters the queue. The recording includes an entrance announcement and then ringback or other announcements while the call is in the queue.

If there is no entrance announcement, it is possible that no recording is started if the call is immediately offered to an agent or if the call is affected by the Size Overflow policy that immediately transfers the call.

5.12.2.7.1.1 Overflow Policy

Calls that are already in or just entering the call queues can be affected by the Overflow policies. The Overflow policies are triggered by the size of the current queue or the time in the current queue. The Overflow policies can be configured with three options when the size or time is exceeded.

- Busy Treatment – The behavior is described in section *5.12.2.4 Busy Treatment*.
- Transfer – The behavior is described in section *5.11 Call Transfer*.
- Ringing – The call is provided with ringing until the call is abandoned. Similar to busy treatment and transfer, an optional announcement can be configured to be played prior to ringback being applied. In this case, the call recording continues until the call is abandoned. The recording contains an optional announcement and then ringback until the user abandons the call, or the recording contains a portion of the above that is played until the user abandons the call.

5.12.2.7.1.2 Bounced

A bounced call is one that has been offered to an agent but the call was not answered by the agent. The call is placed back in the queue at the highest priority. In this scenario, the call recording is stopped when the call is offered to the agent. Since the agent does not take the call, the call recording is started again on behalf of the Call Center, recording an announcement that might be played as part of the Bounce policy and also recording the duration of the call as it is "re-queued", until an agent finally answers.

5.12.2.7.1.3 Stranded

A stranded call is one that is waiting in a queue, but there are no agents logged in to service the queue. The Stranded policy describes how these calls are handled. The options for handling these calls are:

- None – The call remains in the queue and the recording continues until the user abandons the call.
- Busy – The behavior is described in section *5.12.2.4 Busy Treatment*.
- Transfer – The behavior is described in section *5.11 Call Transfer*.
- Night Service – The behavior is described in section *5.12.2.6 Pre-queue Policies*.

- Ringing – The call continues ringing and the recording continues until the caller abandons the call.
- Announcement – An announcement is looped until the user abandons the call. The call recording continues until the user abandons the call.

5.12.2.7.2 Presented to Agent

When the call is presented to the agent, the call recording associated with the Call Center ends when the agent answers the call. The call recording for the agent starts after the agent answers the call and after the whisper announcement is played, if provisioned. The call is only recorded if the answering agent has the Call Recording service active.

In some cases, the Call Center agent might be located on a different Application Server than the Call Center; this is known as a migrated Call Center agent. In this situation, the call recording starts on the new Application Server just as it would as described above; however, the external tracking ID would be new.

5.12.2.7.3 Supervisor Takes Over Call

When a supervisor takes over a call from an agent and if the supervisor has the Call Recording service, then a call recording starts the moment the supervisor takes over the call.

If the supervisor was monitoring and recording the agent's call prior to taking it over, the recording continues.

5.12.2.7.4 Transfer Call to Another Call Center

When a call is transferred to another Call Center, the call recording for the original Call Center ends. If the new Call Center has Call Recording assigned, then a new recording begins.

5.12.3 Route Point

The Route Point service is similar to the Call Center behavior. Route Point queries a third-party server for its queuing decisions, rather than the logic running on the Cisco BroadWorks Application Server. For more information on the Route Point service, see the *Network CTI Integration Feature Description* [11]. The behavior of the following services and recordings are the same as they are for the Call Center service:

- Night Service
- Holiday Service
- Forced Forwarding (CFA)
- Overflow – Size and Time
- Bounced Call

The Route Point has an additional policy and one additional capability not available to the call center. The additional policy is the External System Failure policy. The additional capability is the Outgoing Dial Request. Details of the Call Recording service interactions with the new policy and capability follow.

5.12.3.1 Video Support

A route point can also be configured to play video announcements and treatments for all the same scenarios configured for audio. If video is streamed in any of these scenarios, the video is also recorded along with the audio.

5.12.3.2 End-User Notification

There is no announcement played for the call recording start/stop on behalf of Route Point because the End-User Notification of Recording feature is not applied to the virtual user.

The end-user notification behavior on the Route Point call is based on the call originator or the destination agent.

Calling party with Call Recording in the Always mode

User A dials the Route Point number and connected to the Route Point. The call recording on behalf of User A starts and the Recording Started announcement is played User A.

When the call is routed to the agent User B and User B answers, the call recording continues and the *Recording Started* announcement is replayed to User B only.

Agent with Call Recording in the Always mode

The call recording on behalf of the call receiving agent starts when the agent answers. The *Recording Started* announcement is played to both calling party and receiving agent.

5.12.3.3 Meet-Me Conference

When a call recording user joins the Meet-Me Conference, it is treated as a regular two-party call in which one is the call recording user and the other is the Meet-Me Conference.

Joining party with Call Recording in the Always mode

Assume only User A has the Call Recording service in *Always* mode and the option "*Play Call Recording Start/Stop Announcement*" is enabled.

1) User A makes a call to join the Meet-Me Conference.

   User A is first connected to the Meet-Me IVR and at the meantime the first call recording starts. User A hears the announcement "*This call is being recorded.*" overlapped with the Meet-Me welcome announcement.

2) User A enters the Meet-Me Conference ID and successfully joins the Meet-Me Conference.

   The first call recording stops but the second call recording starts. User A hears the announcement "This call is being recorded." overlapped with the Meet-Me success announcement.

NOTE: When the second call recording starts, User A is still connected to the Meet-Me IVR. By the time User A is really in the conference with other conference parties, the *Recording Started* announcement has been played and the other conference parties miss it.

3) User A hangs up and leaves the Meet-Me Conference.

NOTE: Since the call between User A and the Meet-Me Conference is treated as a regular two-party call and there is no way to tell if User A is tearing down a two-party call or leaving a conference, no *Recording Ended* announcement is played to the remaining parties in the Meet-Me Conference.

5.12.3.4 Voice Mail Recording

A route point that has both the Call Recording and Voice Messaging User services has the ability to record voice message deposits. If the administrator has *Record Voice Messaging* enabled, then the voice message deposits are recorded.

5.12.3.5 External Failure Policy

Since the Route Point service relies on an external server to make routing decisions, there are cases where the external server is unavailable. In these cases, the call is handled by the External Failure policy. This policy can affect new calls that are not yet in the queue as well as those calls already in the queue. When the route point detects the failure of the external server and the External Failure policy is active, the calls already in the queue are transferred to the failover destination. This transfer is throttled to control the number of calls sent to the failover destination. When the call is transferred the call recording stops for that call. The only exception is that any Outgoing Dial Request call is not transferred but is released, and call recording stops at that point. If the call is not in the queue, then the call recording does not start for the route point because the call is immediately sent to the failure destination when the call is received.

5.12.3.6 Outgoing Dial Request

The route point can also initiate an outgoing call to place the user in the route point queue. Once the target has answered the call, the call is placed into the route point queue to be offered to the agent. If the route point has the Call Recording service, the call recording starts when the call enters the queue. From this point onward, the Call Recording service treats this call in the same manner as it does any other queued call.

5.13 Other Services

5.13.1 Answer Confirmation

The call recording starts after the Answer Confirmation service accepts the call.

5.13.2 BroadWorks Anywhere

Calls terminating to a BroadWorks Anywhere subscriber are recorded on behalf of the BroadWorks Anywhere user, even if the call is answered by the BroadWorks Anywhere location.

5.13.3 BroadWorks Mobility

Calls to and from a user with the BroadWorks Mobility service are recorded. This includes calls to and originated from the subscriber's mobile number provisioned in the BroadWorks Mobility service.

5.13.4 Call Forward Always

If the call terminates to a user with the Call Forwarding Always service active, the call is not recorded for the subscriber with Call Forwarding Always active.

5.13.5 Call Pickup

If a user who picks up the call has Call Recording, then the call is recorded.

5.13.6 Call Me Now

All calls that terminate to a Cisco BroadWorks Call Me Now subscriber are recorded after the subscriber answers the call.

5.13.7 Music/Video On Hold

The Music/Video On Hold service allows a user or administrator to configure a music and a video source to be played to callers on held or parked calls. The behavior of call recording when Video On Hold is active is the same as it is for Music On Hold except that the video portion of the call is also recorded in addition to the music.

5.13.8 Push To Talk

If a user with the Call Recording service instantiates a one-way voice path Push-To-Talk call, then that is one recording.

If the other party involved in the one-way voice path Push-To-Talk conversation wants to instantiate voice path in the other direction, then that is another recording.

5.13.9 Remote Office

Calls terminating to a Remote Office subscriber are recorded on behalf of the Remote Office user, even if the call is answered by a Remote Office location.

5.13.10 Sequential Ringing

If the terminating subscriber has Sequential Ringing and Call Recording services, the call is only recorded for the subscriber if that subscriber answers the call.

If one of the sequential ring destinations answers the call, it is not recorded for the sequential ring subscriber.

If the destination that answers the call has the Call Recording service, then the call is recorded for the answering party.

5.13.11 Shared Call Appearance

Calls terminating to a Shared Call Appearance subscriber are recorded on behalf of the Shared Call Appearance user, even if the call is answered by a Shared Call Appearance location.

An alternate Shared Call Appearance location of a subscriber that created an N-Way conference call would not be allowed to bridge into the conference call when the bridging subscriber has the Call Recording service enabled.

5.13.12 Simultaneous Ring

If the terminating subscriber has the Simultaneous Ringing and Call Recording services, the call is only recorded for the subscriber if that subscriber answers the call.

If one of the simultaneous ring destinations answers the call, it is not recorded for the simultaneous ring subscriber.

If the destination that answers the call has the Call Recording service, then the call is recorded for the answering party.

5.13.13 Third-Party Voice Mail Support

The Call Recording service does not record any voice messages left on a third-party voice messaging system.

5.13.14 Emergency Calls

The Call Recording service does not record an emergency call.

6 Provisioning

6.1 Provisioning Flow

Perform the steps in the following sections to provision this feature.

6.1.1 System Level

1) Obtain a license for the Call Recording service.
2) Provision the system recording platform(s) at the CLI.
3) Provision whether the call is allowed to continue in the event of a call recording failure by setting the *continueCallAfterRecordingFailure* system parameter.
4) Provision whether a video call is allowed to continue in the event of a call recording failure by setting *continueCallAfterVideoRecordingFailure* system parameter.
5) Provision whether *on Demand* mode follows the above two parameters by setting *useContinueCallAfterRecordingFailureForOnDemandMode.*
6) Provision whether *On Demand with user Initiated Start mode* follows the parameters in steps 3 and 4 by setting *useContinueCallAfterRecordingFailureForOnDemandUserInitiatedStartMode*.
7) To change the wait time of a response from the server, configure the *maxResponseWaitTimeMilliSeconds* system parameter.
8) Voice Mail Recording:
   - Provision the system-level *schema_version* Call Recording platform parameter to set the version of the Cisco BroadWorks-specific extension metadata to 3.0, if supported by the recording platform.
9) Video Call Support:
   - To determine whether a video call is allowed to continue in the event of a call recording failure, provision the system-level *continueCallAfterVideoRecordingFailure* call recording parameter.
   - To determine whether a call recording for a video call is supported, provision the system-level *supportVideoRec* recording platform parameter.
10) Visual Security Classification:
   - Provision the system-level *schema_version* Call Recording platform parameter to set the version of the Cisco BroadWorks-specific extension metadata to 3.0, if supported by the recording platform.

6.1.2 Enterprise/Service Provider Level

1) Authorize the Call Recording service to the enterprise/service provider.
2) Provision the FAC *Record Call*.
3) Start, Stop, Pause, and Resume.
   - To use a non-default FAC for call recording “start”, modify the FAC at the enterprise/service provider level.
   - To use a non-default FAC for call recording “stop”, modify the FAC at the enterprise/service provider level.

- To use a non-default FAC for call recording "pause", modify the FAC at the enterprise/service provider level.
- To use a non-default FAC for call recording "resume", modify the FAC at the enterprise/service provider level.
- To configure the pause/resume notification mode, set *Pause/Resume Notification* to "None", "Beep", or "Play Announcement".

6.1.3 Group Level

1) Authorize the Call Recording service for the group.
2) Provision the FAC *Record Call*.
3) Start, Stop, Pause, and Resume.
   - To use a non-default FAC for call recording "start", modify the FAC at the group level.
   - To use a non-default FAC for call recording "stop", modify the FAC at the group level.
   - To use a non-default FAC for call recording "pause", modify the FAC at the group level.
   - To use a non-default FAC for call recording "resume", modify the FAC at the group level.
   - To configure the pause/resume notification mode, set *Pause/Resume Notification* to "None", "Beep", or "Play Announcement".
4) If the group uses a different call recording platform, then change the Call Recording platform to use it.

6.1.4 User Level

1) Assign the service to the user or to the following virtual services:
   - Call Centers
   - Route Points
   - Auto Attendant
2) Assign the level of service:
   - To configure the call recording mode, set *Record Call* to "Never", "Always", "Always with Pause/Resume", "On Demand", or "On Demand with User Initiated Start".
   - Set the level of service for virtual services to "Never" or "Always".
3) Start, Stop, Pause, and Resume:
   - To configure the pause/resume notification mode, set *Pause/Resume Notification* to "None", "Beep", or "Play Announcement".
4) End-User Notification:
   - Determine whether the end user receives an announcement notification when a call is recorded, by enabling the *Call Recording Start Stop Announcement* parameter.

- Determine whether the end user receives a repeated warning tone while the call is being recorded by enabling the *Record Call Repeat Warning Tone* parameter.
- Control how often the end user receives the repeated warning tone that the call is being recorded, by provisioning the *Record Call Repeat Warning Tone Timer* parameter.

5) Voice Mail Recording:

   - Provision the user's *Record Voice Mail Messages* Call Recording service parameter to enable the call recording of voice message(s) left for the user in their voice mailbox.

6) Call Control for IP Phones:

   - To enable/disable the Call Recording feature synchronization for shared devices, provision the user's device policies parameter *Enable Call Recording.*

6.2 Service Description and Licensing

6.2.1 BroadWorks Call Recording Service

A new user level licensable service, BroadWorks Call Recording, is created for this feature.

6.2.1.1 Application Server

Official Service Name	License Type	Service Quantity Type
Call Recording	User	UserService

User Service Assignable Condition	Yes/No
Assignable to a real user?	Yes
Assignable to an Auto Attendant?	Yes
Assignable to a call center?	Yes
Assignable to a hunt group?	No
Assignable to an instant conference?	No
Assignable to an instant group call?	No
Assignable to a route point?	Yes
Assignable to a BroadWorks Anywhere?	No
Assignable to a paging group?	No
Allowed in a service pack?	Yes

6.3 Application Server Provisioning Steps

The following section describes the required provisioning for the Application Server.

6.3.1 Service Activation for Release 17.sp4

Activate the service feature as follows.

```
AS_CLI/System/ActivatableFeature> activate 46941
```

6.3.2 Activatable Feature ID and Dependencies

- Activatable Feature ID: 46941.
- Activatable Feature Name: Call Recording.
- Dependencies: FR 140637 *"Enable CDR schema version R17 SP4 for Activatable Features".*

6.3.3 Mid-call Provisioning Changes

It is possible for a user or an administrator to change the recording mode of a user's service while a call is in session. The effects of these transitions are listed in the following table. The first column lists the current recording state. Columns 2 through 4 list the new recording mode and the effect the change has on the recording. This table only applies to calls in progress when the provisioning change is made.

	Always	Always with Pause/Resume	On-Demand	On Demand with User-Initiated Start	Never
Always	There is no effect.	The call recording continues for current call.	The call recording continues for current call.	The call recording continues for current call.	The call recording continues for current call.
Always with Pause/Resume	The call recording with Pause/Resume functionality continues for current call.	There is no effect.	The call recording with Pause/Resume functionality continues for current call.	The call recording with Pause/Resume functionality continues for current call.	The call recording with Pause/Resume functionality continues for current call. If the pause/resume FAC is dialed, it is ignored.
On-Demand	The in-progress call recording is not kept unless the FAC is dialed or the Xsi command or "recordpref:on" is received. When kept, the Pause/Resume functionality becomes applicable to the call recording.	The in-progress call recording is not kept unless the FAC is dialed or the Xsi command or "recordpref:on" is received. When kept, the Pause/Resume functionality becomes applicable to the call recording.	There is no effect.	The in-progress call recording is not kept unless the FAC is dialed or if the Xsi command or "recordpref:on" is received. When kept, the Pause/Resume functionality becomes applicable to the call recording.	The in-progress call recording is not kept unless the Xsi command or "recordpref:on" is received. If the FAC is dialed, it is ignored. When kept, the Pause/Resume functionality becomes applicable to the call recording. If the pause/resume FAC is dialed, it is ignored.
On-Demand with User-Initiated Start	The call recording does not begin unless the user starts it. When started, Pause/Resume/ Stop functionality is applicable to the call recording.	The call recording does not begin unless the user starts it. When started, Pause/Resume/Stop functionality is applicable to the call recording.	The call recording does not begin unless the user starts it. When started, Pause/Resume/ Stop functionality is applicable to the call recording.	There is no effect.	The call recording does not begin unless the user starts it. If the FAC is dialed, it is ignored. When started, Pause/Resume/Stop functionality is applicable to the call recording. If the pause/resume/stop FAC is dialed, it is ignored.
Never	The call recording does not start for the current call.	The call recording does not start for the current call.	The call recording does not start for the current call.	The call recording does not start for the current call.	There is no effect.

6.3.3.1 Video Support

- If the *supportVideoRecording* parameter is changed, it does not affect the state of any call recording sessions already in progress. The changes are only applied to new call recording sessions once the change takes effect on the Application Server.
- If the *continueCallAfterVideoRecordingFailure* parameter is changed, it does not affect the state of any call recording sessions already in progress. The changes are only applied to new call recording sessions once the change takes effect on the Application Server.

6.3.3.2 End-User Notification

It is possible for a user or an administrator to change the call recording end-user notification parameters of the Call Recording service while a call is in session. These changes do not affect calls in progress. Once a call starts, the call recording end-user notification parameters are set for the life of the call. Only calls that a user originates or receives after the provisioning change are affected by the new settings.

6.3.3.3 Controls for IP Phones

The call recording user's provisioned recording mode may be modified from Cisco BroadWorks through the web portal or the Xtended Services Interface.

If the user's recording mode is modified after the initial subscription notification is sent to the SIP endpoint device, the newly modified recording mode is indicated in the *CallRecordingModeEvent* of a subsequent NOTIFY, as shown in *Figure 65*.

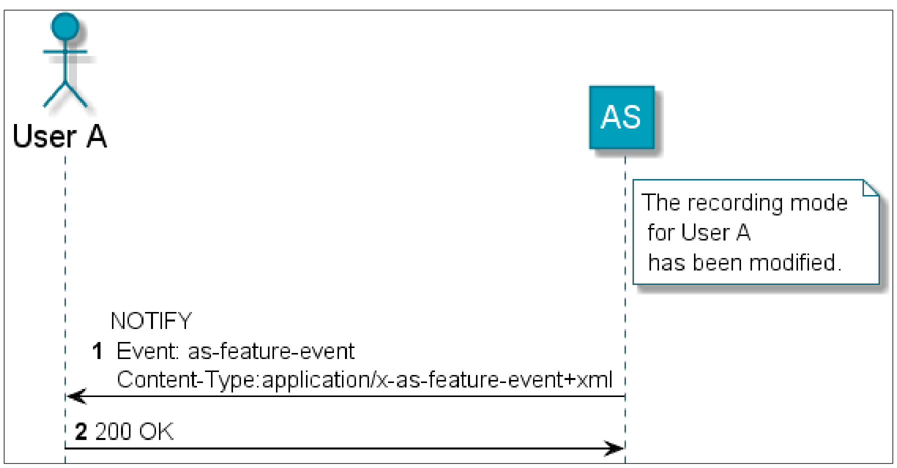

Figure 2 NOTIFY Sent When Provisioned Recording Mode Modified

It is possible that the user's recording mode is modified while the user has an active call (or active calls). During mid-call provisioning changes to a user's recording mode, the newly provisioned recording mode does **not** take effect on the Cisco BroadWorks Application Server for the user's active call(s). Their active call(s) continue to use the recording mode that was provisioned at the time the call was established. The newly provisioned recording mode only takes effect for the user's subsequent calls.

NOTE 1 (on recording-aware client behavior): When the provisioned recording mode is modified, the Cisco BroadWorks Application Server sends a NOTIFY to the recording-aware UA that contains the user's newly modified recording mode. The NOTIFY may arrive while the user is still on an active call (or active calls), in which case, the recording-aware UA must continue to display the appropriate recording controls based on the original recording mode at the time the call was established. For subsequent new calls for the user, the recording-aware UA must display the appropriate controls based on the newly modified recording mode.

NOTE 2 (on recording-aware client behavior): There is currently no mechanism for the recording-aware device to modify the recording mode via the SIP interface.

6.3.4 Recording Platform Provisioning

The Call Recording service can be authorized down to the user level without any restrictions. However, because recording platform(s) must be provisioned for a recording to take place, turning on the feature for a specific user, either by selecting *Always* or *On-Demand* is possible only if a recording platform is assigned to the user via its group.

If at least one recording platform is defined on the system, the group inherits the system default recording platform when the service is authorized to the group.

Once a recording platform is assigned to a group, it is not possible to clear the selection other than assigning another platform to the group. Only when "unauthorizing" the service to the group, its assigned recording platform is cleared.

The system default recording platform cannot be deleted unless it is the unique one defined, provided it is not assigned to any group.

In addition, see section *5.4.2 Load Balancing Using 302 Redirection Support*.

6.3.5 Add Call Recording Platform(s)

Use the *add* command to add Call Recording Platforms to the Application Server.

```
AS_CLI/Service/CallRecording/add [name] [netAddress] [transportType]
[mediaStream] [schemaVersion][supportVideoRecording] description
[description label] port [port]
```

Where:

Name	The name of the recording platform (1-80 characters)
netAddress	This is the fully qualified domain name (FQDN), host, or IP address of the recording platform.
transportType	This is the SIP interface type ("UDP", "TCP", or "Unspecified").
mediaStream	This is the type of media stream defined either as "dual" or "single" stream.
schemaVersion	The version of the Cisco BroadWorks-specific extension metadata that the recording platform understands (1.0, 2.0, 3.0).

supportVideoRecording	This parameter determines if video call recording is supported by the recording platform (false, true).
description value	This is the description of this recording platform.
Port	This is the address port of the recording platform (integer, 1 to 65535). **NOTE**: This port is optional and clearable. The add and clear commands are updated for SRV Support. For more information, see section *6.3.6 Provisioning SRV for Call Recording Platform(s)*.

Example:

```
AS_CLI/Service/CallRecording> add platformA RD_FQDN UDP single 3.0
false description RecordingDeviceFQDN port 5060

...Done
```

If an FQDN is specified as the *netAddress*, then the INVITE to the 3PCR platform has the FQDN in the request URI. However, the Application Server resolves the address and distribute invites to the resolved addresses. How the distribution is done depends on the Domain Name System (DNS) resolution type of the FQDN.

6.3.6 Provisioning SRV for Call Recording Platform(s)

Release 22.0 introduces support for SRV for FQDN resolution of the call recording platform. For more information, see the *SRV Support for Call Recording Platforms Feature Description* [27].

6.3.6.1 SRV Support Implementation

SRV records can be used to resolve the Call Recording platform's FQDN.

> **NOTE**: For a SRV lookup to be provisioned, there should be **NO** port defined for the Call Recording platform.

The FQDN for the Call Recording platform(s) are resolved on a per-call basis at runtime. The server list for a given platform creates a match each time an attempt is made to connect to the3PCR platform. If the local DNS cache has expired, then a new DNS query is sent to resolve the address. Otherwise, the cached value is used. The cache can be manually cleared from the DNS CLI level. In addition, if the *namedefs* file is updated and the reload command is used to refresh the information, then the next call after the reload uses the information from the revised *namedefs* file.

If the list of Call Recording servers for a platform is exhausted, then the Call Recording service treats the platform as failed, the appropriate alarm is raised, and the call fails or continues based on the setting of the *continueCallAfterRecordingFailure* for the *Always* modes For the *On Demand* and *On Demand with User Initiated Start* modes, whether the call continues or follows the *continueCallAfterRecordingFailure* parameters is dependent on the *useContinueCallAfterRecordingFailureForOnDemandMode* and the *useContinueCallAfterRecordingFailureForOnDemandUserInitiatedStartMode* system parameters. If either of these parameters is "true" then the following description for the always mode calls also applies to any on demand mode with the parameter set to "true". If the mode is either *Always* or *Always with Pause/Resume,* the call does not immediately fail. If the 3PCR platform server is out of service, it waits for the *maxResponseWaitTimeMilliseconds* to detect that each server is out of service. This means that a 3PCR platform that has three servers takes (3 x *maxResponseWaitTimeMilliseconds)* seconds before the call is taken down. Prior to the call being taken down, users have voice path. For example, if your *maxResponseWaitTimeMilliseconds* is configured for three seconds and the FQDN for the 3PCR platform resolves into two servers, both of which have just become unreachable, a call would have six seconds of voice path before being taken down. In the case of a call transfer, the call does not have voice for the six seconds while waiting for the connection to fail.

The addresses for the Call Recording platform are resolved in the order of the following precedence:

1) SRV records defined in *namedefs*
2) SRV records from the DNS server
3) A records from *namedefs*
4) A records from the DNS server

The order of the SRV records is calculated using the priority and then the weight, as defined in *RFC 2782* [26].

The Application Server builds an ordered list of the Call Recording platforms using either SRV or A records. This list determines the order in which the actual Call Recording platform servers are attempted.

If a Call Recording platform definition that did not previously have a port defined is modified to include a port, then the *Server* list is rebuilt for the affected Call Recording platform using only A record results. If a port is defined, no attempt is made to resolve the FQDN using SRV records.

6.3.6.2 Parameters Removed Due to SRV Support

Since the resolution of the addresses for the Call Recording platforms are now done at runtime, the following parameters no longer have any effect:

- *refreshPeriodSeconds*
- *maxConsecutiveFailures*

The Application Server continues to attempt to connect to servers that have gone down for maintenance or for which the Application Server has lost connectivity.

Prior to the Release 22.0 SRV Support feature, these two parameters would allow a server that we had failed to connect to for the value defined in *maxConsecutiveFailures* to be skipped for the number of seconds defined in *refreshPeriodSeconds*. This feature removes this functionality in favor of resolution of the network address at the time the connection to the Call Recording platform is made.

6.3.6.3 Alarm Changes Due to SRV Support

The medium alarm, *bwCallRecordingPlatformError*, was only raised prior to Release 22.0 when the Call Recording platform sent an error response or released a call unexpectedly.

This feature now raises a medium alarm if we are unable to connect a server. If the entire Call Recording platform is unreachable, then a high alarm is raised. However, the medium alarm is only raised for the first server that we cannot reach in the Call Recording platform.

The medium alarm is cleared when the server that the alarm was raised for is successfully connected or if the high alarm is raised. The resolved IP address and the FQDN that defines the platform are both included in the alarm to allow for investigation of the connection failure.

6.3.6.4 SRV Example

The following example demonstrates using SRV to resolve the address for the Call Recording platform.

1) Add the following entries to the DNS server or *namedefs* file used by the Application Server:
   - _sip._udp.cr.broadsoft.com 86400 IN SRV 2 5 5060 cr3.broadsoft.com
   - _sip._udp.cr.broadsoft.com 86400 IN SRV 0 70 5060 cr1.broadsoft.com
   - _sip._udp.cr.broadsoft.com 86400 IN SRV 0 30 5060 cr2.broadsoft.com
   - cr1.broadsoft.com IN A 10.16.12.25
   - cr2.broadsoft.com IN A 10.16.18.89
   - cr3.broadsoft.com IN A 10.16.24.7
2) Define a Call Recording platform.

```
         Name            Net Address     Port      Transport Type   media
Stream   Schema Version   Support Video Rec   Description

=========================================================================
=======================
    srvplatform     cr.broadsoft.com                   Unspecified
dual                        3.0                         false
```

3) Assign the new Call Recording platform to Group A.
4) Enable call recording for User 1 in Group A.
5) Place a call to User 1.
6) The Call Recording platforms are ordered by priority and then by weight.
7) The Call Recording platform selected by the Application Server is cr1.broadsoft.com or cr2.broadsoft.com. The one selected first varies based on the weight. Weight gives precedence to the larger weight values but attempts to balance the platforms. In the example, cr1 is first 70 percent of the time and cr2 30 percent of the time. This example assumes that cr1 is first.
8) If cr1.broadsoft.com is not available, the Application Server route advances to cr2.broadsoft.com and then to cr3.broadsoft.com.

6.3.7 Set Default Call Recording Platform

If there are multiple Call Recording platforms configured in the Application Server then the administrator must indicate which platform is Default.

Use the *set* method to indicate the default platform.

```
AS_CLI/Service/CallRecording/set [name] systemDefault true
```

Where [name] is the name value of a Call Recording platform.

Example:

```
AS_CLI/Service/CallRecording> set platformB systemDefault true
```

The *get* command allows for verification of available platforms and default setting.

```
AS_CLI/Service/CallRecording> get
```

Example:

```
System default Call Recording platform = platformB
```

```
Name        Net Address    Port    Transport Type Media Stream  Description
========================================================================
platformA   RD_FQDN        5070               UDP          dual  dell-960
platformB   192.168.8.34   5060               TCP        single  ibm-x552
platformC   RD_HOST        5071       Unspecified          dual  dell-990
```

6.3.8 BroadWorks Call Recording Service Administration Configuration

The following menus have been modified to add links to the new *BroadWorks Recording Services* page:

- *ServiceProvider/Enterprise → Resources → Services*
- *Group → Resources → Services*
- *User → Call Control*

The following pages have been added to support the new BroadWorks Call Recording service:

- *Service Provider/Enterprise → Utilities → FACs*
- *Group → Utilities → FACs*
- *User → Call Control → BroadWorks Call Recording (administrator view)*
- User → Call Control → BroadWorks Call Recording (user view)

6.3.9 Enterprise/Service Provider Level Configuration

The following sections describe the changes affecting the Service Provider and Enterprise level configuration.

6.3.9.1 Enterprise/Service Provider Administrator Services

System administrators can authorize the BroadWorks Call Recording service for enterprises / service providers. Authorizing this service allows the service provider/enterprise administrators to authorize their groups for this service.

This page is accessed by enterprise/service provider administrators. A new User service row for the Call Recording service is visible when the service is licensed and authorized to the enterprise/service provider.

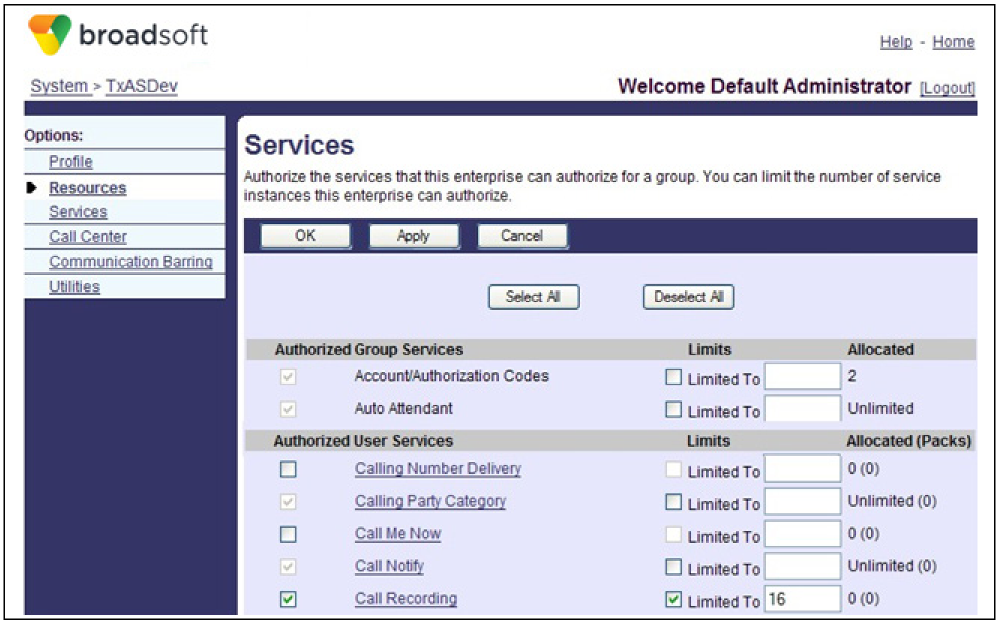

Figure 3 Enterprise/Service Provider → Resources → Services Page

6.3.9.2 Service Provider/Enterprise FACs

The *Enterprise/Service Provider → Utilities → Feature Access Codes* page is accessed by enterprise/service provider administrators. A new FAC *Record Call* is visible when the service is licensed and authorized to the enterprise/service provider.

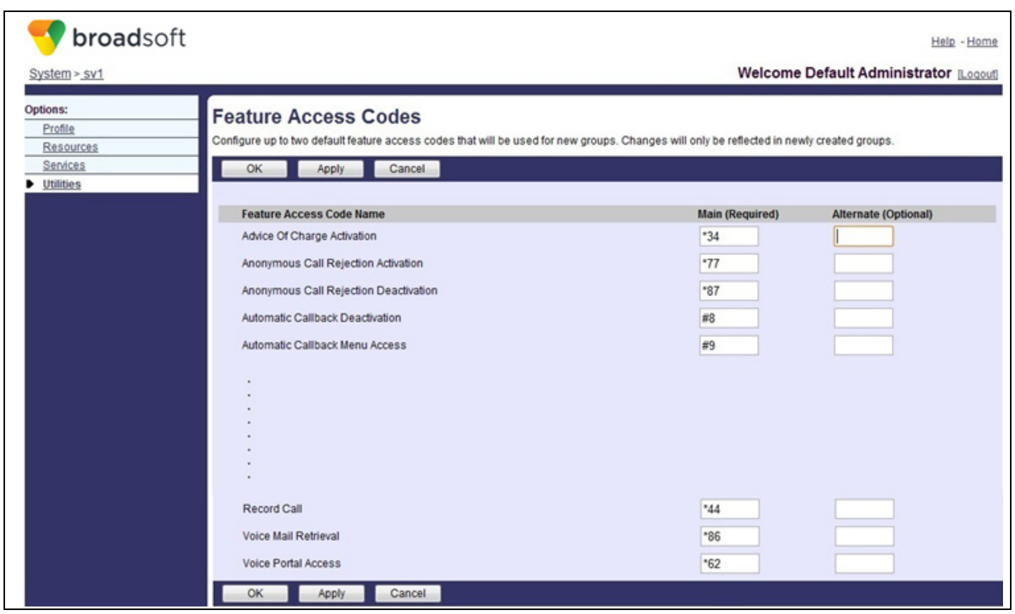

Figure 4 Enterprise/Service Provider → Utilities → Feature Access Codes Page

6.3.10 Group Level Configuration

Service provider/enterprise administrators can authorize the BroadWorks Call Recording service for their groups.

6.3.10.1 Group Administrator Services

The *Group → Resources → Services* page is accessed by group administrators. A new User service row for the Call Recording service is visible when the service is licensed and authorized to the group.

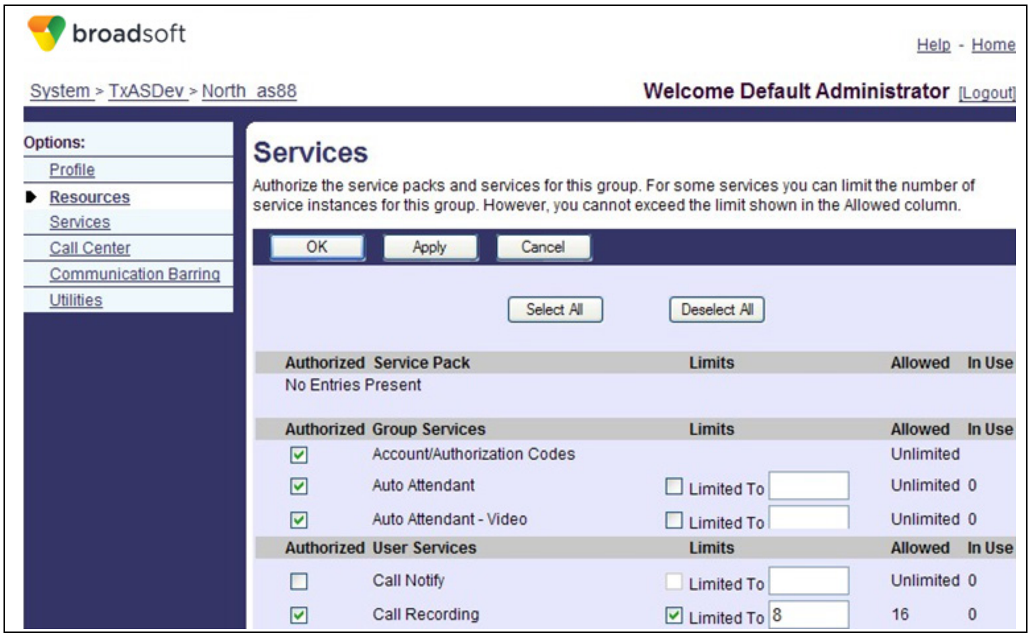

Figure 5 Group → Resources → Services Page

6.3.10.2 Group FACs

The *Group → Utilities → Feature Access Codes* page is accessed by the system, enterprise/service provider, and group administrators. A new FAC Record Call is visible when the service is licensed and authorized to the group.

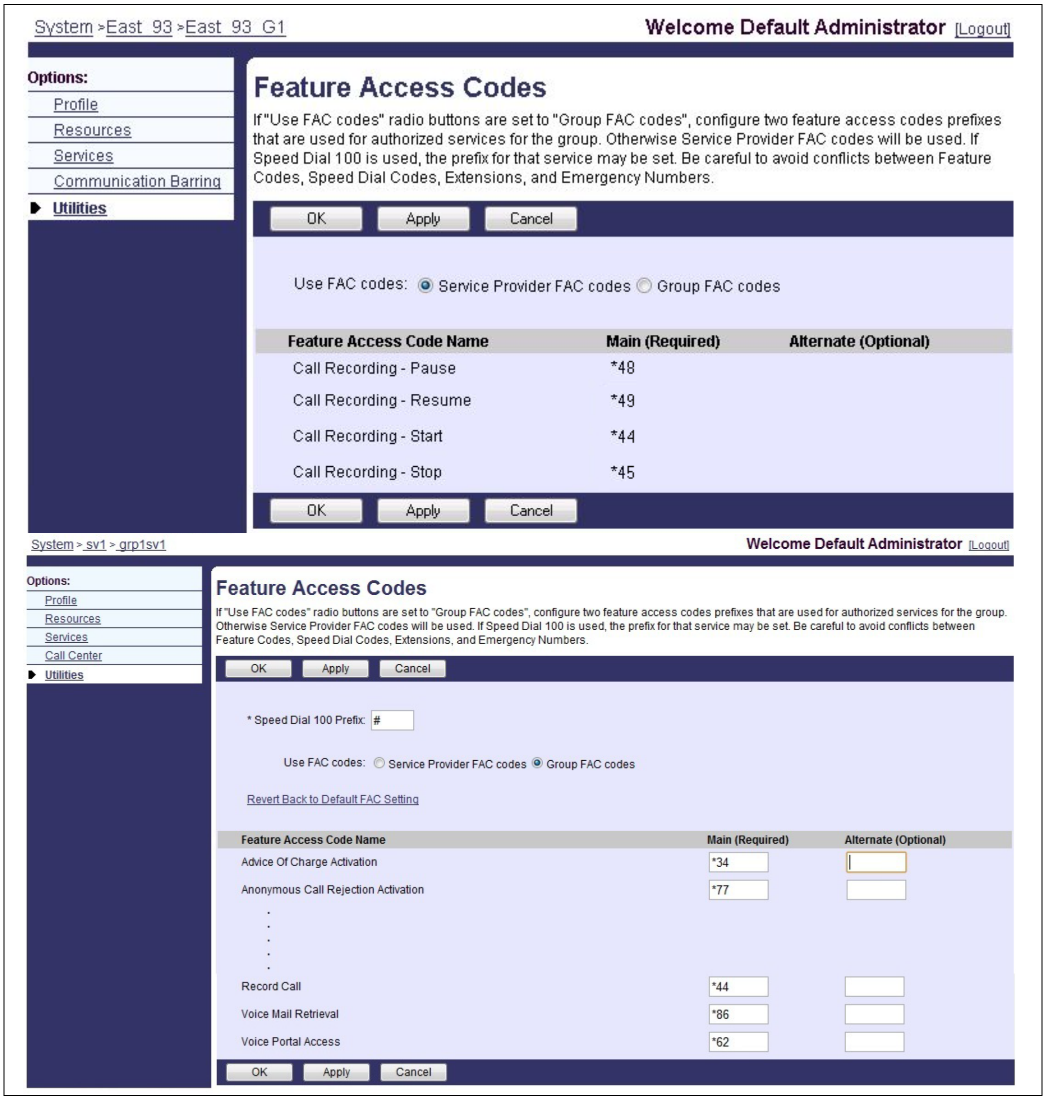

Figure 6 Group → Utilities → Feature Access Codes Page

6.3.10.3 Group Resources Menu

The *Group → Resources* menu is accessed by the system, enterprise/service provider, and group administrators. A link to configure the group's recording platform is added if the Call Recording service is authorized to the group.

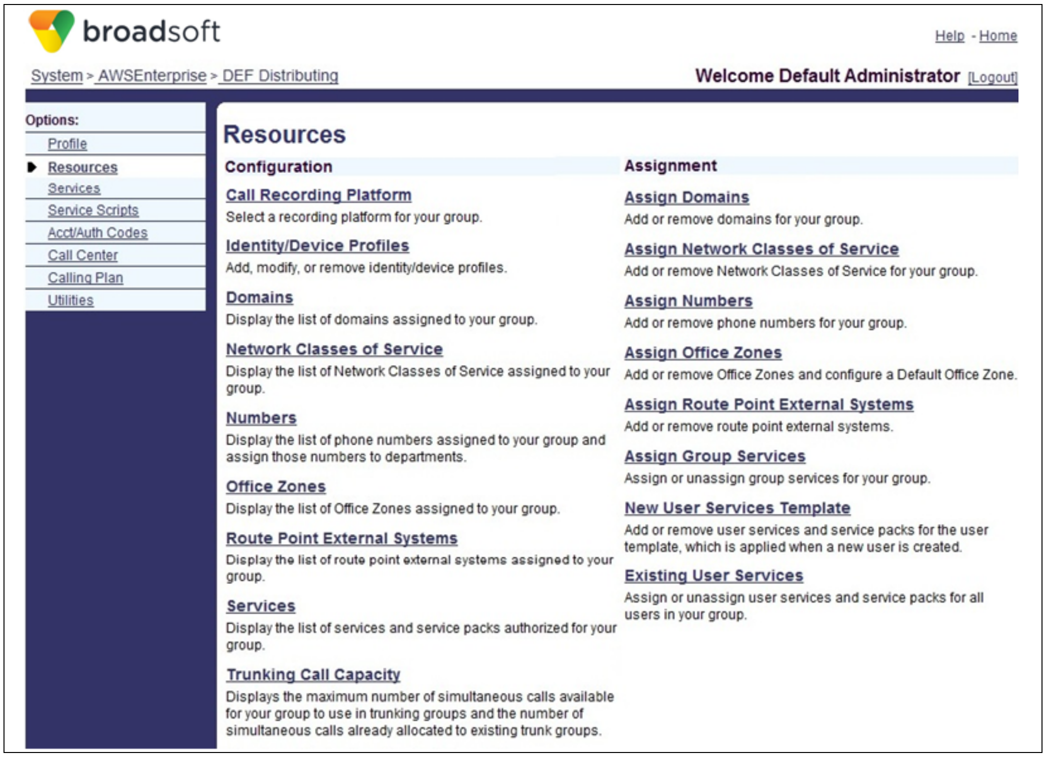

Figure 7 Group → Resources Menu Page

6.3.10.4 Group Resources Call Recording Platform Page

The *Group → Resources → Call Recording Platform* page is accessed by the system, enterprise/service provider, and group administrators. A recording platform can be assigned for the group. The "None" option is only shown when the group does not yet have a recording platform assigned. Once a recording platform is assigned to a group, it is not possible to clear the selection other than assigning another platform to the group. Only when unauthorizing the service to the group, is its assigned recording platform cleared.

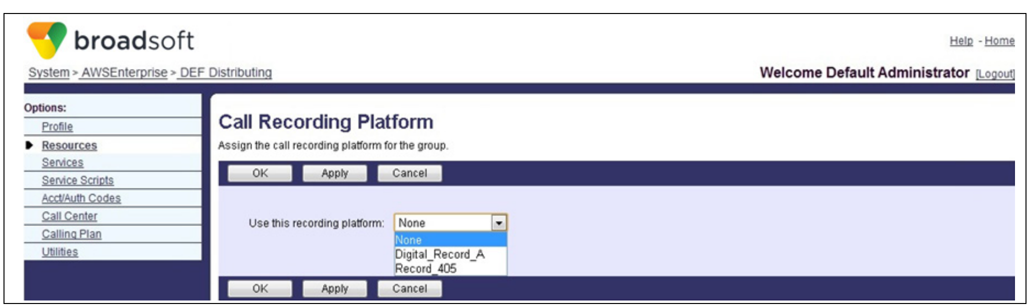

Figure 8 Group → Resources → Call Recording Platform Page

6.3.11 User Level Configuration

Group administrators can assign the BroadWorks Call Recording service to a user.

6.3.11.1 Assign Call Recording Service

Group administrators must assign the BroadWorks Call Recording service to a user from the user's *Assign Services* page.

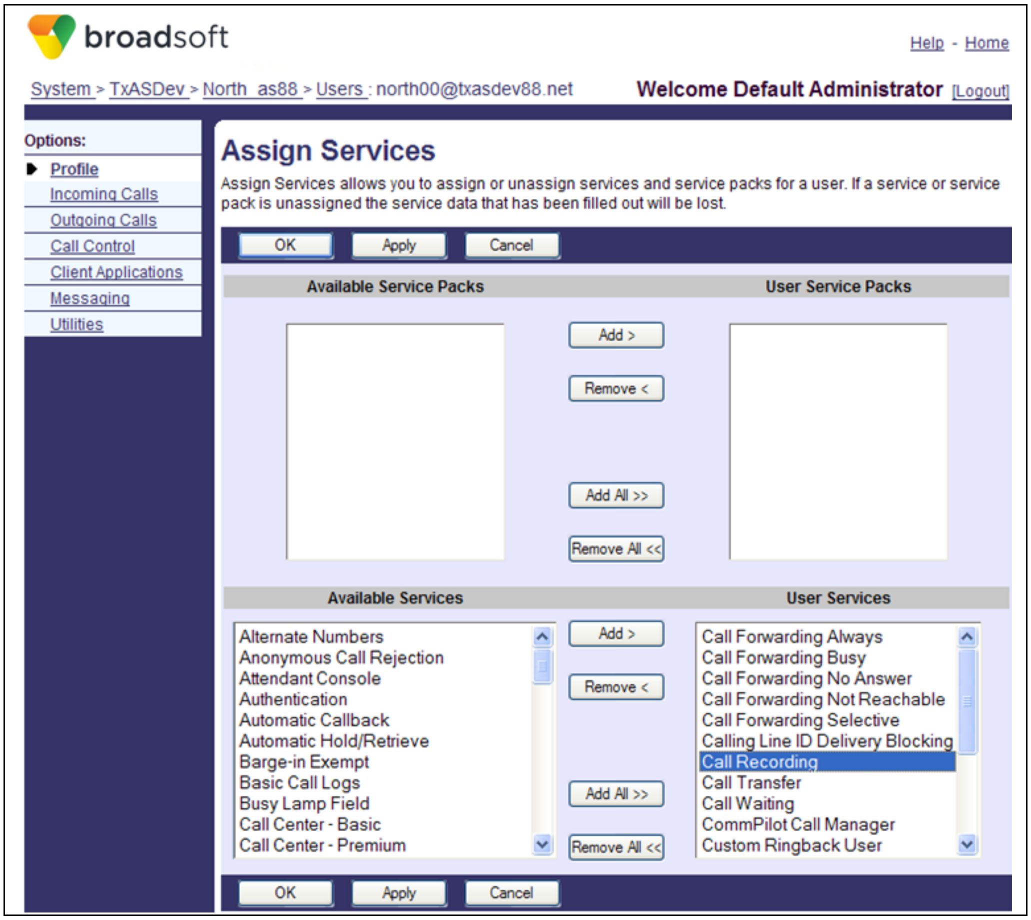

Figure 9 User → Assign Services Page

6.3.11.2 Call Control for IP Phones

A new setting is added under Multiple User Shared Lines option to enable/disable the Call Recording feature synchronization for shared devices on the *User → Profile → Device* Policies page.

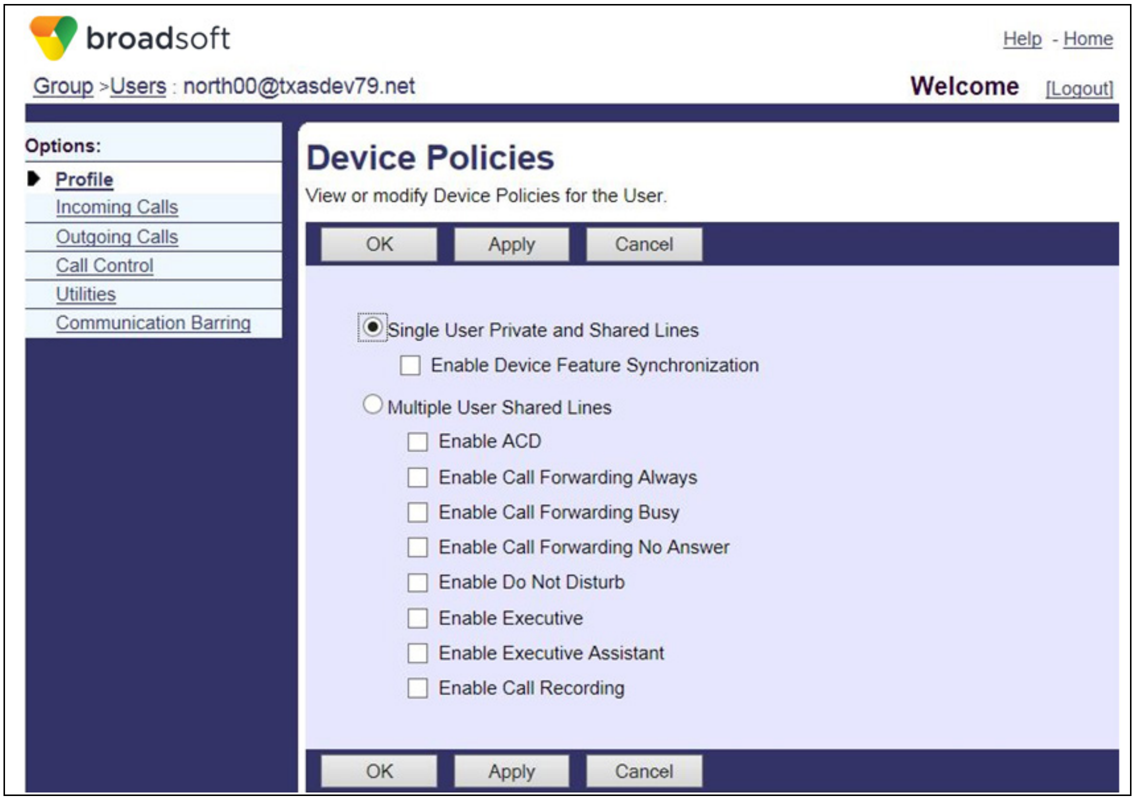

Figure 10 User → Profile → Device Policies Page

6.3.11.3 User Call Control Menu

The *User → Call Contro*l menu page is accessed by the system, enterprise/service provider, and group administrators, along with the user. A link to configure the Call Recording service is visible when the service is assigned to the user. "On" is displayed in the link when either *Always, Always with Pause/Resume, On-Demand,* or *On-Demand with User Initiated Start* is selected.

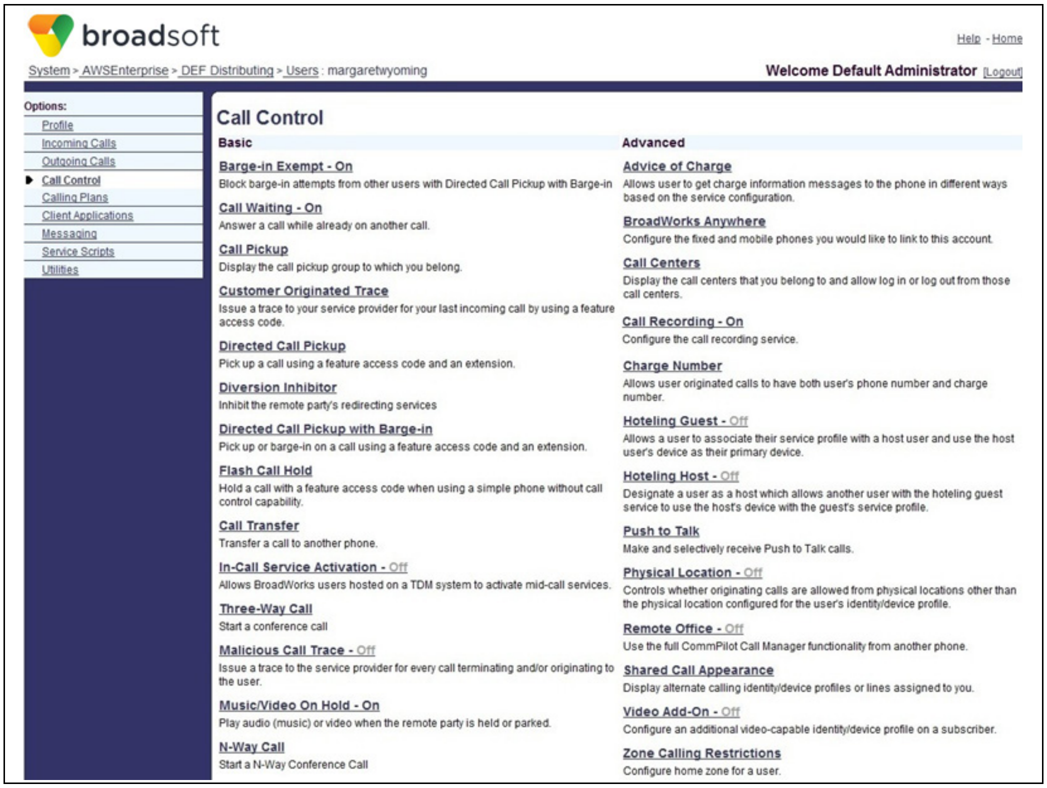

Figure 11 User → Call Control Menu Page

6.3.11.4 User Call Recording Page

The User → Call Control → Call Recording page is be accessed by the system, enterprise/service provider, and group administrators, along with the user. The Call Recording service, when provisioned for the user, can be turned on by selecting *Always, Always with Pause/Resume, On-Demand,* or *On-Demand with User Initiated Start* or can be turned off by selecting *Never*.

Figure 12 User → Call Control → Call Recording Page

6.3.11.5 Service Instance Call Control Menu

The *User → Call Control* menu page is accessed by the system, enterprise/service provider, and group administrators, along with the Auto Attendant, Call Center, and Route Point service instances. A link to configure the Call Recording service is visible when the service is assigned to those service instances.

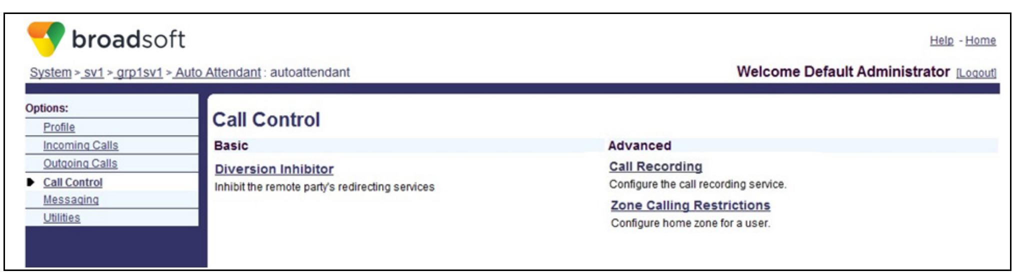

Figure 13 User → Call Control Service Instance Page

6.3.11.6 Service Instance Call Recording Page

The *User → Call Control → Call Recording* page is accessed by the system, enterprise/service provider, and group administrators. The Call Recording service, when provisioned for those service instances, can be turned on by selecting *Always* or can be turned off by selecting *Never*.

Figure 14 User → Call Control → Call Recording Service Instance Page

7 Accounting Management

7.1 Generation of Accounting Records

There are two Call Recording service extensions.

The *first* Call Recording service extension captures details regarding the FAC invocation of the *Always with Pause/Resume, On-Demand*, and *On-Demand with User Initiated Start* recording. This service extension is named “Call Recording Invocation”, and it includes the same type of information that other FAC service extensions capture:

- *Invocation Time* – This field captures the time at which the service was invoked during the call. The invocation time is shown using the Coordinated Universal Time (UTC)/Greenwich Mean Time (GMT) time zone.
- *FAC Result* – This field captures the result (“Success” or “Failure”) of dialing a feature access code. The value “Success” means the feature was invoked and processed successfully.

The *second* Call Recording service extension captures details regarding the recording, such as the user’s recording mode, the destination where the recording is saved, and the success/failure of the recording. This service extension is named “Call Recording”, and it contains the following information:

- Call Recording Trigger – Indicates the user’s recording mode for this call; the values for this field are “Always”, “Always-Pause-Resume”, “On-Demand”, or “On-Demand-User-Start”.

NOTE: Although “Never” can be configured for a user’s Call Recording service, it is not necessary to capture *Call Recording* fields for calls that are never recorded, and therefore, the value “Never” is not applicable for this field.

- *Call Recording Destination* – Indicates the identity/address of the 3PCR platform where the media stream is sent. This destination can be in the format of an IP address, host name, or FQDN.
- *Call Recording Result* – Indicates the status of the recorded media; the values for this field can be “successful”, “failed”, or “successful but not kept”.
- If the Call Recording Trigger is “Always”, “Always-Pause-Resume”, or “On-Demand-User-Start”, then the Call Recording Result can be “successful” or “failed”.
- If the Call Recording Trigger is “Always”, “Always-Pause-Resume”, “On-Demand”, or “On-Demand-User-Start”, then the Call Recording Result can be “successful”, “failed”, or “successful but not kept”.
- A “successful” result means that there were no errors while the Application Server made the necessary connections on the Media Server and connected the media to the 3PCR platform for recording.
- A “failed” result means that there was a general error encountered by the Application Server. This may include connection/communication issues with the Media Server and/or 3PCR platform, or it may be an internal error on the Application Server.

- A “successful but not kept” result means that the recording was successful, but the user did not request the recording to be saved to the 3PCR platform. Calls made by users with the “On-Demand” setting are automatically streamed to the 3PCR platform by default. However, for the recording to be saved, the user must request it to be saved via FAC or Xtended Services Interface.

The initial snapshot of the Call Recording service extension captures the Call Recording Trigger and the Call Recording Destination. The initial snapshot may also capture the Call Recording Result if the user’s calls are configured to always record.

- For users who are configured with *On-Demand* recording, the snapshot of the Call Recording Result, along with the Call Recording Invocation service extension, is captured upon receiving the request to save an *On-Demand* recording, and these are written to the final call detail record (CDR) when the call is terminated.

If the 3PCR platform has a mechanism to request that a recording be kept in the *On-Demand* cases and this mechanism is used, the *Call Recording* field is reported as “successful but not kept” by the accounting record.

7.1.1 Start, Stop, Pause, Resume

There are two new values added to *Call Recording Trigger* which are:

- “always-pause-resume”
- “on-demand-user-start”

When one of the new modes is present in the *Call Recording Trigger* field, the *Call Recording Result* field is populated as follows:

- If the *Call Recording Trigger* is “always-pause-resume”*,* then the *Call Recording Result* may be “successful” or “failed”.
- If the *Call Recording Trigger* is “on-demand-user-initiated-start”*,* then the *Call Recording Result* may be “successful” or “failed”.

For each of the four new FAC invocations that are introduced by this feature, new service extensions are added. The new service extensions are:

- On-Demand-Start Call Recording
- On-Demand-Stop Call Recording
- Pause Call Recording
- Resume Call Recording

7.1.2 Video Support

Call Recording of Video does not make any changes to the accounting records created by the Call Recording feature. Accounting records are created to record calls with video in a manner similar to that of audio-only calls. The accounting records created for call recording with audio and video do not differ from the records created for audio-only calls.

7.1.3 End-User Notification

End-User Notification of Call Recording does not make any changes to the accounting records created by the Call Recording feature.

7.1.4 Voice Mail Recording

Voice Mail Call Recording does not make any changes to the accounting records created by the Call Recording feature.

7.1.5 Controls for IP Phones

Controls for IP phones do not make any changes to the accounting records created by the Call Recording feature.

7.2 Impact to Accounting Fields (CDR)

Field name:	*CallRecordingInvocationTime*
XML tag name:	invocationTime
CSV column number in normal/long duration CDRs:	315
CSV column number in failover CDRs:	Not applicable
Radius ID:	315
Radius dictionary:	BWAS-Call-Recording-Invocation-Time
SQL database column:	CallRecordingInvocationTime
Module:	Centrex, within the “Call Recording Invocation” service extension
Optional or mandatory:	Optional
Description:	This service extension field indicates the time at which the service was invoked during the call. The invocation time is shown using the UTC/GMT time zone.
Maximum string length:	18 characters
Value can contain non-standard ASCII characters?	No
Example data:	`20110421215831.471`

Field name:	*CallRecordingFACResult*
XML tag name:	facResult
CSV column number in normal/long duration CDRs:	316
CSV column number in failover CDRs:	Not applicable
Radius ID:	316
Radius dictionary:	BWAS-Call-Recording-FAC-Result
SQL database column:	CallRecordingFACResult
Module:	Centrex, within the “Call Recording Invocation” service extension
Optional or mandatory:	Optional
Description:	This service extension field captures the result (“Success” or “Failure”) of dialing a FAC. The value “Success” means the feature was invoked and processed successfully.
Maximum string length:	7 characters
Value can contain non-standard ASCII characters?	No
Example data:	`Success`

Field name:	*CallRecordingTrigger*

XML tag name:	recordingTrigger
CSV column number in normal/long duration CDRs:	317
CSV column number in failover CDRs:	Not applicable
Radius ID:	317
Radius dictionary:	BWAS-Call-Recording-Trigger
SQL database column:	CallRecordingTrigger
Module:	Centrex, within the “Call Recording” service extension
Optional or mandatory:	Optional
Description:	This service extension field indicates the user’s recording mode for this call; the values for this field are “always”, “always-pause-resume”, “on-demand”, or “on-demand-user-start”.
Maximum string length:	20 characters
Value can contain non-standard ASCII characters?	No
Example data:	on-demand-user-start

Field name:	*CallRecordingDestination*
XML tag name:	recordingDestination
CSV column number in normal/long duration CDRs:	318
CSV column number in failover CDRs:	Not applicable
Radius ID:	318
Radius dictionary:	BWAS-Call-Recording-Destination
SQL database column:	CallRecordingDestination
Module:	Centrex, within the “Call Recording” service extension
Optional or mandatory:	Optional
Description:	This service extension field indicates the identity/address of the 3PCR platform where the media stream is sent. This destination could be in the format of an IP address, host name, or FQDN.
Maximum string length:	161 characters
Value can contain non-standard ASCII characters?	No
Example data:	`10.16.150.10`

Field name:	*CallRecordingResult*
XML tag name:	recordingResult
CSV column number in normal/long duration CDRs:	319
CSV column number in failover CDRs:	Not applicable
Radius ID:	319
Radius dictionary:	BWAS-Call-Recording-Result

SQL database column:	CallRecordingResult
Module:	Centrex, within the "Call Recording" service extension
Optional or mandatory:	Optional
Description:	This service extension field indicates the status of the recorded media; the values for this field can be "successful", "failed", or "successful but not kept". If the *recordingTrigger* is "always", then the *recordingResult* can be "successful" or "failed". If the *recordingTrigger* is "on-demand", then the *recordingResult* can be "successful", "failed", or "successful but not kept". A "successful" result means that there were no errors while the Application Server made the necessary connections on the Media Server and connected the media to the 3PCR platform for recording. A "failed" result means that there was a general error encountered by the Application Server. This may include connection/communication issues with the Media Server and/or 3PCR platform, or it may be an internal error on the Application Server. A "successful but not kept" result means that the recording was successful, but the user did not request the recording to be saved to the 3PCR platform. Calls made by users with the "on-demand" setting are automatically streamed to the 3PCR platform by default; however, for the recording to be saved, the user must request it to be saved via FAC or Xtended Services Interface.
Maximum string length:	23 characters
Value can contain non-standard ASCII characters?	No
Example data:	`successful but not kept`

Release 20.0

Field name:	*CallRecordingPause.invocationTime*
XML tag name:	invocationTime
CSV column number in normal/long duration CDRs:	368
CSV column number in failover CDRs:	Not applicable
Radius ID:	368
Radius dictionary:	BWAS-Call-Recording-Pause-Invocation-Time
SQL database column:	CallRecordingPause.invocationTime
Module:	Centrex
Optional or mandatory:	Optional
Description:	This service extension field indicates the time at which the service was invoked during the call. The invocation time is shown using the UTC/GMT time zone.
Maximum string length:	18 characters
Value can contain non-standard ASCII characters?	No
Example data:	`20110421215831.471`

Field name:	*CallRecordingPause.facResult*
XML tag name:	facResult
CSV column number in normal/long duration CDRs:	369
CSV column number in failover CDRs:	Not applicable
Radius ID:	369
Radius dictionary:	BWAS-Call-Recording-Pause-Fac-Result
SQL database column:	CallRecordingPause.facResult
Module:	Centrex
Optional or mandatory:	Optional
Description:	This field captures the result ("Success" or "Failure") of dialing a FAC. The value "Success" means the feature was invoked and processed successfully.
Maximum string length:	7 characters
Value can contain non-standard ASCII characters?	No
Example data:	`Success`

Field name:	*CallRecordingResume.invocationTime*
XML tag name:	invocationTime
CSV column number in normal/long duration CDRs:	370
CSV column number in failover CDRs:	Not applicable
Radius ID:	370
Radius dictionary:	BWAS-Call-Recording-Resume-Invocation-Time
SQL database column:	CallRecordingResume.invocationTime
Module:	Centrex
Optional or mandatory:	Optional
Description:	This service extension field indicates the time at which the service was invoked during the call. The invocation time is shown using the UTC/GMT time zone.
Maximum string length:	18 characters
Value can contain non-standard ASCII characters?	No
Example data:	`20110421215831.471`

Field name:	*CallRecordingResume.facResult*
XML tag name:	facResult
CSV column number in normal/long duration CDRs:	371
CSV column number in failover CDRs:	Not applicable
Radius ID:	371
Radius dictionary:	BWAS-Call-Recording-Resume-Fac-Result

SQL database column:	CallRecordingResume.facResult
Module:	Centrex
Optional or mandatory:	Optional
Description:	This field captures the result (“Success” or “Failure”) of dialing a FAC. The value “Success” means the feature was invoked and processed successfully.
Maximum string length:	7 characters
Value can contain non-standard ASCII characters?	No
Example data:	`Success`

Field name:	*CallRecordingStart.invocationTime*
XML tag name:	invocationTime
CSV column number in normal/long duration CDRs:	364
CSV column number in failover CDRs:	Not applicable
Radius ID:	364
Radius dictionary:	BWAS-Call-Recording-Start-Invocation-Time
SQL database column:	CallRecordingStart.invocationTime
Module:	Centrex
Optional or mandatory:	Optional
Description:	This service extension field indicates the time at which the service was invoked during the call. The invocation time is shown using the UTC/GMT time zone.
Maximum string length:	18 characters
Value can contain non-standard ASCII characters?	No
Example data:	`20110421215831.471`

Field name:	*CallRecordingStart.facResult*
XML tag name:	facResult
CSV column number in normal/long duration CDRs:	365
CSV column number in failover CDRs:	Not applicable
Radius ID:	365
Radius dictionary:	BWAS-Call-Recording-Start-Fac-Result
SQL database column:	CallRecordingStart.facResult
Module:	Centrex
Optional or mandatory:	Optional
Description:	This field captures the result (“Success” or “Failure”) of dialing a FAC. The value “Success” means the feature was invoked and processed successfully.
Maximum string length:	7 characters
Value can contain non-standard ASCII characters?	No

Example data:	`Success`

The following example is for an originating user who has the Call Recording service configured as "on-demand". Although the call was streamed to the 3PCR platform successfully, the user did not issue an *On-Demand* request to save it.


```
<cdrData>
  <headerModule>
    <recordId>
      <eventCounter>0000000006</eventCounter>
      <systemId>DEFAULT</systemId>
      <date>20110415224516.273</date>
       <systemTimeZone>1-050000</systemTimeZone>
    </recordId>
    <serviceProvider>TxASDev</serviceProvider>
    <type>Normal</type>
  </headerModule>
  <basicModule>
    <userNumber>+19726996500</userNumber>
    <direction>Originating</direction>
    <asCallType>Group</asCallType>
    <callingNumber>+19726996500</callingNumber>
    <callingPresentationIndicator>Public</callingPresentationIndicator>
    <dialedDigits>501</dialedDigits>
    <calledNumber>+19726996501</calledNumber>
    <startTime>20110415224516.273</startTime>
    <userTimeZone>1-050000</userTimeZone>
    <localCallId>15:0</localCallId>
    <remoteCallId>19:0</remoteCallId>
    <answerIndicator>Yes</answerIndicator>
    <answerTime>20110415224521.850</answerTime>
    <releaseTime>20110415224606.711</releaseTime>
    <terminationCause>016</terminationCause>
    <callCategory>private</callCategory>
    <chargeIndicator>n</chargeIndicator>
    <releasingParty>remote</releasingParty>
    <userId>north00@txasdev96.net</userId>
    <otherPartyName>john1 north</otherPartyName>

<otherPartyNamePresentationIndicator>Public</otherPartyNamePresentationIndicator
>
  </basicModule>
  <centrexModule>
    <group>North_as96</group>
    <locationList>
      <locationInformation>
        <location>9726996500@txasdev96.net</location>
        <locationType>Primary Device</locationType>
      </locationInformation>
    </locationList>
    <locationUsage>44.671</locationUsage>
    <serviceExtensionList>
      <serviceExtension>
        <serviceName>Call Recording</serviceName>
        <recordingTrigger>on-demand</recordingTrigger>
        <recordingDestination>10.16.150.10</recordingDestination>
        <recordingResult>successful but not kept</recordingResult>
      </serviceExtension>
    </serviceExtensionList>
  </centrexModule>
  <ipModule>
    <route>Group</route>
    <codec>PCMA/8000</codec>
    <accessDeviceAddress>10.16.150.100</accessDeviceAddress>
    <accessCallID>bddc78a4ca9dbef4@10.16.150.100</accessCallID>
    <codecUsage>44.666</codecUsage>
  </ipModule>
```

```
</cdrData>
```


The following example is for an originating user who has the Call Recording service configured as *On-Demand* and the user has dialed the FAC *before* originating the call.


```
<cdrData>
  <headerModule>
    <recordId>
      <eventCounter>0000000006</eventCounter>
      <systemId>DEFAULT</systemId>
      <date>20110421215831.351</date>
      <systemTimeZone>1-050000</systemTimeZone>
    </recordId>
    <serviceProvider>TxASDev</serviceProvider>
    <type>Normal</type>
  </headerModule>
  <basicModule>
    <userNumber>+19726996500</userNumber>
    <direction>Originating</direction>
    <asCallType>Internal</asCallType>
    <callingNumber>+19726996500</callingNumber>
    <callingPresentationIndicator>Public</callingPresentationIndicator>
    <dialedDigits>*44501</dialedDigits>
    <calledNumber>*44501</calledNumber>
    <startTime>20110421215831.351</startTime>
    <userTimeZone>1-050000</userTimeZone>
    <localCallId>143:0</localCallId>
    <answerIndicator>No</answerIndicator>
    <releaseTime>20110421215836.131</releaseTime>
    <terminationCause>016</terminationCause>
    <chargeIndicator>n</chargeIndicator>
    <releasingParty>remote</releasingParty>
    <userId>north00@txasdev96.net</userId>
  </basicModule>
  <centrexModule>
    <group>North_as96</group>
    <serviceExtensionList>
      <serviceExtension>
        <serviceName>Call Recording Invocation</serviceName>
        <invocationTime>20110421215831.471</invocationTime>
        <facResult>Success</facResult>
      </serviceExtension>
      <serviceExtension>
        <serviceName>Call Recording</serviceName>
        <recordingTrigger>on-demand</recordingTrigger>
        <recordingDestination>10.16.150.10</recordingDestination>
        <recordingResult>successful</recordingResult>
      </serviceExtension>
    </serviceExtensionList>
    <locationList>
      <locationInformation>
        <location>9726996500@txasdev96.net</location>
        <locationType>Primary Device</locationType>
      </locationInformation>
    </locationList>
    <locationUsage>4.620</locationUsage>
  </centrexModule>
  <ipModule>
    <codec>PCMU/8000</codec>
    <accessDeviceAddress>10.16.150.100</accessDeviceAddress>
    <accessCallID>12654a64eca68ffd@10.16.150.100</accessCallID>
    <codecUsage>4.620</codecUsage>
  </ipModule>
</cdrData>
```

The following example is for an originating user who has the Call Recording service configured as *On-Demand* and who dials the FAC *during* the middle of the call to save the recording. This scenario results in two CDRs for the originating user: The first CDR captures the call to the called party and the second CDR captures the FAC invocation during the middle of the call.

```
<cdrData>
  <headerModule>
    <recordId>
      <eventCounter>0000000011</eventCounter>
      <systemId>DEFAULT</systemId>
      <date>20110421222036.366</date>
      <systemTimeZone>1-050000</systemTimeZone>
    </recordId>
    <serviceProvider>TxASDev</serviceProvider>
    <type>Normal</type>
  </headerModule>
  <basicModule>
    <userNumber>+19726996500</userNumber>
    <direction>Originating</direction>
    <asCallType>Group</asCallType>
    <callingNumber>+19726996500</callingNumber>
    <callingPresentationIndicator>Public</callingPresentationIndicator>
    <dialedDigits>501</dialedDigits>
    <calledNumber>+19726996501</calledNumber>
    <startTime>20110421222036.366</startTime>
    <userTimeZone>1-050000</userTimeZone>
    <localCallId>163:0</localCallId>
    <remoteCallId>167:0</remoteCallId>
    <answerIndicator>Yes</answerIndicator>
    <answerTime>20110421222038.247</answerTime>
    <releaseTime>20110421222126.501</releaseTime>
    <terminationCause>016</terminationCause>
    <callCategory>private</callCategory>
    <chargeIndicator>n</chargeIndicator>
    <releasingParty>local</releasingParty>
    <userId>north00@txasdev96.net</userId>
    <otherPartyName>john1 north</otherPartyName>

<otherPartyNamePresentationIndicator>Public</otherPartyNamePresentationIndicator
>
  </basicModule>
  <centrexModule>
    <group>North_as96</group>
    <locationList>
      <locationInformation>
        <location>9726996500@txasdev96.net</location>
        <locationType>Primary Device</locationType>
      </locationInformation>
    </locationList>
    <locationUsage>48.220</locationUsage>
    <serviceExtensionList>
      <serviceExtension>
        <serviceName>Call Recording</serviceName>
        <recordingTrigger>on-demand</recordingTrigger>
        <recordingDestination>10.16.150.10</recordingDestination>
        <recordingResult>successful</recordingResult>
      </serviceExtension>
    </serviceExtensionList>
  </centrexModule>
  <ipModule>
    <route>Group</route>
    <codec>PCMA/8000</codec>
    <accessDeviceAddress>10.16.150.100</accessDeviceAddress>
    <accessCallID>d72021e725012758@10.16.150.100</accessCallID>
    <codecUsage>48.218</codecUsage>
  </ipModule>
</cdrData>
```

```
<cdrData>
  <headerModule>
    <recordId>
      <eventCounter>0000000012</eventCounter>
      <systemId>DEFAULT</systemId>
      <date>20110421222100.247</date>
      <systemTimeZone>1-050000</systemTimeZone>
    </recordId>
    <serviceProvider>TxASDev</serviceProvider>
    <type>Normal</type>
  </headerModule>
  <basicModule>
    <userNumber>+19726996500</userNumber>
    <direction>Originating</direction>
    <asCallType>Internal</asCallType>
    <callingNumber>+19726996500</callingNumber>
    <callingPresentationIndicator>Public</callingPresentationIndicator>
    <dialedDigits>*44</dialedDigits>
    <calledNumber>*44</calledNumber>
    <startTime>20110421222100.247</startTime>
    <userTimeZone>1-050000</userTimeZone>
    <localCallId>163:1</localCallId>
    <answerIndicator>No</answerIndicator>
    <releaseTime>20110421222126.501</releaseTime>
    <terminationCause>016</terminationCause>
    <chargeIndicator>n</chargeIndicator>
    <releasingParty>remote</releasingParty>
    <userId>north00@txasdev96.net</userId>
  </basicModule>
  <centrexModule>
    <group>North_as96</group>
    <locationList>
      <locationInformation>
        <location>9726996500@txasdev96.net</location>
        <locationType>Primary Device</locationType>
      </locationInformation>
    </locationList>
    <locationUsage>48.220</locationUsage>
    <serviceExtensionList>
      <serviceExtension>
        <serviceName>Call Recording Invocation</serviceName>
        <invocationTime>20110421222100.247</invocationTime>
        <facResult>Success</facResult>
      </serviceExtension>
    </serviceExtensionList>
  </centrexModule>
  <ipModule>
    <route>Group</route>
    <codec>PCMA/8000</codec>
    <accessDeviceAddress>10.16.150.100</accessDeviceAddress>
    <accessCallID>d72021e725012758@10.16.150.100</accessCallID>
    <codecUsage>48.218</codecUsage>
  </ipModule>
</cdrData>
```


The following example is for a terminating user who has the Call Recording service configured as “always”. As soon as the terminating user answers the call, the call is automatically recorded.


```
<cdrData>
  <headerModule>
    <recordId>
      <eventCounter>0000000012</eventCounter>
      <systemId>DEFAULT</systemId>
      <date>20110420164828.616</date>
      <systemTimeZone>1-050000</systemTimeZone>
    </recordId>
    <serviceProvider>TxASDev</serviceProvider>
    <type>Normal</type>
```

```
  </headerModule>
  <basicModule>
    <userNumber>+19726996502</userNumber>
    <direction>Terminating</direction>
    <asCallType>Group</asCallType>
    <callingNumber>+19726996501</callingNumber>
    <callingPresentationIndicator>Public</callingPresentationIndicator>
    <calledNumber>+19726996502</calledNumber>
    <startTime>20110420164828.616</startTime>
    <userTimeZone>1-050000</userTimeZone>
    <localCallId>319:0</localCallId>
    <remoteCallId>315:0</remoteCallId>
    <answerIndicator>Yes</answerIndicator>
    <answerTime>20110420164833.245</answerTime>
    <releaseTime>20110420171439.116</releaseTime>
    <terminationCause>016</terminationCause>
    <callCategory>private</callCategory>
    <chargeIndicator>n</chargeIndicator>
    <releasingParty>remote</releasingParty>
    <userId>north02@txasdev96.net</userId>
    <otherPartyName>john1 north</otherPartyName>

<otherPartyNamePresentationIndicator>Public</otherPartyNamePresentationIndicator
>
    <clidPermitted>Yes</clidPermitted>
    <namePermitted>Yes</namePermitted>
  </basicModule>
  <centrexModule>
    <group>North_as96</group>
    <locationList>
      <locationInformation>
        <location>9726996502@txasdev96.net</location>
        <locationType>Primary Device</locationType>
      </locationInformation>
    </locationList>
    <locationUsage>1565.830</locationUsage>
    <serviceExtensionList>
      <serviceExtension>
        <serviceName>Call Recording</serviceName>
        <recordingTrigger>always</recordingTrigger>
        <recordingDestination>10.16.150.10</recordingDestination>
        <recordingResult>successful</recordingResult>
      </serviceExtension>
    </serviceExtensionList>
  </centrexModule>
  <ipModule>
    <route>Group</route>
    <codec>PCMA/8000</codec>
    <accessDeviceAddress>10.16.150.2</accessDeviceAddress>
    <accessCallID>BW114828655200411-1111068752@10.16.150.2</accessCallID>
    <codecUsage>1565.821</codecUsage>
  </ipModule>
</cdrData>
```

The following example is for an originating user who has the Call Recording service configured as “On Demand User Initiated Start” and who dials the FAC during the middle of the call to start the recording. This scenario results in two CDRs for the originating user. The first CDR captures the start of the call recording and the second CDR captures the FAC invocation during the middle of the call.

```
<cdrData>
  <headerModule>
    <recordId>
      <eventCounter>0000000011</eventCounter>
      <systemId>DEFAULT</systemId>
      <date>20110421222036.366</date>
      <systemTimeZone>1-050000</systemTimeZone>
```

```
    </recordId>
    <serviceProvider>TxASDev</serviceProvider>
    <type>Normal</type>
  </headerModule>
  <basicModule>
    <userNumber>+19726996500</userNumber>
    <direction>Originating</direction>
    <asCallType>Group</asCallType>
    <callingNumber>+19726996500</callingNumber>
    <callingPresentationIndicator>Public</callingPresentationIndicator>
    <dialedDigits>501</dialedDigits>
    <calledNumber>+19726996501</calledNumber>
    <startTime>20110421222036.366</startTime>
    <userTimeZone>1-050000</userTimeZone>
    <localCallId>163:0</localCallId>
    <remoteCallId>167:0</remoteCallId>
    <answerIndicator>Yes</answerIndicator>
    <answerTime>20110421222038.247</answerTime>
    <releaseTime>20110421222126.501</releaseTime>
    <terminationCause>016</terminationCause>
    <callCategory>private</callCategory>
    <chargeIndicator>n</chargeIndicator>
    <releasingParty>local</releasingParty>
    <userId>north00@txasdev96.net</userId>
    <otherPartyName>john1 north</otherPartyName>

<otherPartyNamePresentationIndicator>Public</otherPartyNamePresentationIndicator
>
  </basicModule>
  <centrexModule>
    <group>North_as96</group>
    <locationList>
      <locationInformation>
        <location>9726996500@txasdev96.net</location>
        <locationType>Primary Device</locationType>
      </locationInformation>
    </locationList>
    <locationUsage>48.220</locationUsage>
    <serviceExtensionList>
      <serviceExtension>
        <serviceName>Call Recording</serviceName>
        <recordingTrigger>on-demand-user-start</recordingTrigger>
        <recordingDestination>10.16.150.10:5070</recordingDestination>
        <recordingResult>successful</recordingResult>
      </serviceExtension>
    </serviceExtensionList>
  </centrexModule>
  <ipModule>
    <route>Group</route>
    <codec>PCMA/8000</codec>
    <accessDeviceAddress>10.16.150.100</accessDeviceAddress>
    <accessCallID>d72021e725012758@10.16.150.100</accessCallID>
    <codecUsage>48.218</codecUsage>
  </ipModule>
</cdrData>

<cdrData>
  <headerModule>
    <recordId>
      <eventCounter>0000000012</eventCounter>
      <systemId>DEFAULT</systemId>
      <date>20110421222100.247</date>
      <systemTimeZone>1-050000</systemTimeZone>
    </recordId>
    <serviceProvider>TxASDev</serviceProvider>
    <type>Normal</type>
  </headerModule>
  <basicModule>
    <userNumber>+19726996500</userNumber>
    <direction>Originating</direction>
```

```
    <asCallType>Internal</asCallType>
    <callingNumber>+19726996500</callingNumber>
    <callingPresentationIndicator>Public</callingPresentationIndicator>
    <dialedDigits>*44</dialedDigits>
    <calledNumber>*44</calledNumber>
    <startTime>20110421222100.247</startTime>
    <userTimeZone>1-050000</userTimeZone>
    <localCallId>163:1</localCallId>
    <answerIndicator>No</answerIndicator>
    <releaseTime>20110421222126.501</releaseTime>
    <terminationCause>016</terminationCause>
    <chargeIndicator>n</chargeIndicator>
    <releasingParty>remote</releasingParty>
    <userId>north00@txasdev96.net</userId>
  </basicModule>
  <centrexModule>
    <group>North_as96</group>
    <relatedCallId>163:0</relatedCallId>
    <relatedCallIdReason>Call Recording</relatedCallIdReason>
    <locationList>
      <locationInformation>
        <location>9726996500@txasdev96.net</location>
        <locationType>Primary Device</locationType>
      </locationInformation>
    </locationList>
    <locationUsage>48.220</locationUsage>
    <serviceExtensionList>
      <serviceExtension>
        <serviceName>On-Demand-Start Call Recording</serviceName>
        <invocationTime>20110421222100.247</invocationTime>
        <facResult>Success</facResult>
      </serviceExtension>
    </serviceExtensionList>
  </centrexModule>
  <ipModule>
    <route>Group</route>
    <codec>PCMA/8000</codec>
    <accessDeviceAddress>10.16.150.100</accessDeviceAddress>
    <accessCallID>d72021e725012758@10.16.150.100</accessCallID>
    <codecUsage>48.218</codecUsage>
  </ipModule>
</cdrData>
```


The following example is a continuation of the previous example. In this case, the user decides to stop the call recording during the call. The resulting CDR captures the FAC invocation to stop the call recording prior to the end of the call.


```
<cdrData>
  <headerModule>
    <recordId>
      <eventCounter>0000000012</eventCounter>
      <systemId>DEFAULT</systemId>
      <date>20110421222100.247</date>
      <systemTimeZone>1-050000</systemTimeZone>
    </recordId>
    <serviceProvider>TxASDev</serviceProvider>
    <type>Normal</type>
  </headerModule>
  <basicModule>
    <userNumber>+19726996500</userNumber>
    <direction>Originating</direction>
    <asCallType>Internal</asCallType>
    <callingNumber>+19726996500</callingNumber>
    <callingPresentationIndicator>Public</callingPresentationIndicator>
    <dialedDigits>*45</dialedDigits>
    <calledNumber>*45</calledNumber>
    <startTime>20110421222150.021</startTime>
    <userTimeZone>1-050000</userTimeZone>
    <localCallId>163:1</localCallId>
```

```
    <answerIndicator>No</answerIndicator>
    <releaseTime>20110421222170.501</releaseTime>
    <terminationCause>016</terminationCause>
    <chargeIndicator>n</chargeIndicator>
    <releasingParty>remote</releasingParty>
    <userId>north00@txasdev96.net</userId>
  </basicModule>
  <centrexModule>
    <group>North_as96</group>
    <relatedCallId>163:0</relatedCallId>
    <relatedCallIdReason>Call Recording</relatedCallIdReason>
    <locationList>
      <locationInformation>
        <location>9726996500@txasdev96.net</location>
        <locationType>Primary Device</locationType>
      </locationInformation>
    </locationList>
    <locationUsage>48.220</locationUsage>
    <serviceExtensionList>
      <serviceExtension>
        <serviceName>On-Demand-Stop Call Recording</serviceName>
        <invocationTime>20110421222150.021</invocationTime>
        <facResult>Success</facResult>
      </serviceExtension>
    </serviceExtensionList>
  </centrexModule>
  <ipModule>
    <route>Group</route>
    <codec>PCMA/8000</codec>
    <accessDeviceAddress>10.16.150.100</accessDeviceAddress>
    <accessCallID>d72021e725012758@10.16.150.100</accessCallID>
    <codecUsage>48.218</codecUsage>
  </ipModule>
</cdrData>
```

The following example is for an originating user who has the Call Recording service configured as “always-pause-resume” and who dials the pause FAC during the middle of the call to pause the recording. This scenario results in two CDRs for the originating user. The first CDR captures the call to the called party and the second CDR captures the FAC invocation during the middle of the call.

```
<cdrData>
  <headerModule>
    <recordId>
      <eventCounter>0000000011</eventCounter>
      <systemId>DEFAULT</systemId>
      <date>20110421222036.366</date>
      <systemTimeZone>1-050000</systemTimeZone>
    </recordId>
    <serviceProvider>TxASDev</serviceProvider>
    <type>Normal</type>
  </headerModule>
  <basicModule>
    <userNumber>+19726996500</userNumber>
    <direction>Originating</direction>
    <asCallType>Group</asCallType>
    <callingNumber>+19726996500</callingNumber>
    <callingPresentationIndicator>Public</callingPresentationIndicator>
    <dialedDigits>501</dialedDigits>
    <calledNumber>+19726996501</calledNumber>
    <startTime>20110421222036.366</startTime>
    <userTimeZone>1-050000</userTimeZone>
    <localCallId>163:0</localCallId>
    <remoteCallId>167:0</remoteCallId>
    <answerIndicator>Yes</answerIndicator>
    <answerTime>20110421222038.247</answerTime>
    <releaseTime>20110421222126.501</releaseTime>
    <terminationCause>016</terminationCause>
```

```
    <callCategory>private</callCategory>
    <chargeIndicator>n</chargeIndicator>
    <releasingParty>local</releasingParty>
    <userId>north00@txasdev96.net</userId>
    <otherPartyName>john1 north</otherPartyName>

<otherPartyNamePresentationIndicator>Public</otherPartyNamePresentationIndicator
>
  </basicModule>
  <centrexModule>
    <group>North_as96</group>
    <locationList>
      <locationInformation>
        <location>9726996500@txasdev96.net</location>
        <locationType>Primary Device</locationType>
      </locationInformation>
    </locationList>
    <locationUsage>48.220</locationUsage>
    <serviceExtensionList>
      <serviceExtension>
        <serviceName>Call Recording</serviceName>
        <recordingTrigger>always-pause-resume</recordingTrigger>
        <recordingDestination>10.16.150.10:5070</recordingDestination>
        <recordingResult>successful</recordingResult>
      </serviceExtension>
    </serviceExtensionList>
  </centrexModule>
  <ipModule>
    <route>Group</route>
    <codec>PCMA/8000</codec>
    <accessDeviceAddress>10.16.150.100</accessDeviceAddress>
    <accessCallID>d72021e725012758@10.16.150.100</accessCallID>
    <codecUsage>48.218</codecUsage>
  </ipModule>
</cdrData>

<cdrData>
  <headerModule>
    <recordId>
      <eventCounter>0000000012</eventCounter>
      <systemId>DEFAULT</systemId>
      <date>20110421222100.247</date>
      <systemTimeZone>1-050000</systemTimeZone>
    </recordId>
    <serviceProvider>TxASDev</serviceProvider>
    <type>Normal</type>
  </headerModule>
  <basicModule>
    <userNumber>+19726996500</userNumber>
    <direction>Originating</direction>
    <asCallType>Internal</asCallType>
    <callingNumber>+19726996500</callingNumber>
    <callingPresentationIndicator>Public</callingPresentationIndicator>
    <dialedDigits>*48</dialedDigits>
    <calledNumber>*48</calledNumber>
    <startTime>20110421222100.247</startTime>
    <userTimeZone>1-050000</userTimeZone>
    <localCallId>163:1</localCallId>
    <answerIndicator>No</answerIndicator>
    <releaseTime>20110421222126.501</releaseTime>
    <terminationCause>016</terminationCause>
    <chargeIndicator>n</chargeIndicator>
    <releasingParty>remote</releasingParty>
    <userId>north00@txasdev96.net</userId>
  </basicModule>
  <centrexModule>
    <group>North_as96</group>
    <relatedCallId>163:0</relatedCallId>
    <relatedCallIdReason>Call Recording</relatedCallIdReason>
    <locationList>
```

```
      <locationInformation>
        <location>9726996500@txasdev96.net</location>
        <locationType>Primary Device</locationType>
      </locationInformation>
    </locationList>
    <locationUsage>48.220</locationUsage>
    <serviceExtensionList>
      <serviceExtension>
        <serviceName>Pause Call Recording</serviceName>
        <invocationTime>20110421222100.247</invocationTime>
        <facResult>Success</facResult>
      </serviceExtension>
    </serviceExtensionList>
  </centrexModule>
  <ipModule>
    <route>Group</route>
    <codec>PCMA/8000</codec>
    <accessDeviceAddress>10.16.150.100</accessDeviceAddress>
    <accessCallID>d72021e725012758@10.16.150.100</accessCallID>
    <codecUsage>48.218</codecUsage>
  </ipModule>
</cdrData>
```


The following example is a continuation of the previous example. In this scenario, the user decides to resume the paused call recording. The CDR shows the FAC call to resume the call recording.


```
<cdrData>
  <headerModule>
    <recordId>
      <eventCounter>0000000012</eventCounter>
      <systemId>DEFAULT</systemId>
      <date>20110421222100.247</date>
      <systemTimeZone>1-050000</systemTimeZone>
    </recordId>
    <serviceProvider>TxASDev</serviceProvider>
    <type>Normal</type>
  </headerModule>
  <basicModule>
    <userNumber>+19726996500</userNumber>
    <direction>Originating</direction>
    <asCallType>Internal</asCallType>
    <callingNumber>+19726996500</callingNumber>
    <callingPresentationIndicator>Public</callingPresentationIndicator>
    <dialedDigits>*49</dialedDigits>
    <calledNumber>*49</calledNumber>
    <startTime>20110421222150.021</startTime>
    <userTimeZone>1-050000</userTimeZone>
    <localCallId>163:1</localCallId>
    <answerIndicator>No</answerIndicator>
    <releaseTime>20110421222170.501</releaseTime>
    <terminationCause>016</terminationCause>
    <chargeIndicator>n</chargeIndicator>
    <releasingParty>remote</releasingParty>
    <userId>north00@txasdev96.net</userId>
  </basicModule>
  <centrexModule>
    <group>North_as96</group>
    <relatedCallId>163:0</relatedCallId>
    <relatedCallIdReason>Call Recording</relatedCallIdReason>
    <locationList>
      <locationInformation>
        <location>9726996500@txasdev96.net</location>
        <locationType>Primary Device</locationType>
      </locationInformation>
    </locationList>
    <locationUsage>48.220</locationUsage>
    <serviceExtensionList>
      <serviceExtension>
```

```
        <serviceName>Resume Call Recording</serviceName>
        <invocationTime>20110421222150.021</invocationTime>
        <facResult>Success</facResult>
      </serviceExtension>
    </serviceExtensionList>
  </centrexModule>
  <ipModule>
    <route>Group</route>
    <codec>PCMA/8000</codec>
    <accessDeviceAddress>10.16.150.100</accessDeviceAddress>
    <accessCallID>d72021e725012758@10.16.150.100</accessCallID>
    <codecUsage>48.218</codecUsage>
  </ipModule>
</cdrData>
```

The following example is for a terminating user who has the Call Recording service configured as “always-pause-resume”. As soon as the terminating user answers the call, the call is automatically recorded.

```
<cdrData>
  <headerModule>
    <recordId>
      <eventCounter>0000000012</eventCounter>
      <systemId>DEFAULT</systemId>
      <date>20110420164828.616</date>
      <systemTimeZone>1-050000</systemTimeZone>
    </recordId>
    <serviceProvider>TxASDev</serviceProvider>
    <type>Normal</type>
  </headerModule>
  <basicModule>
    <userNumber>+19726996502</userNumber>
    <direction>Terminating</direction>
    <asCallType>Group</asCallType>
    <callingNumber>+19726996501</callingNumber>
    <callingPresentationIndicator>Public</callingPresentationIndicator>
    <calledNumber>+19726996502</calledNumber>
    <startTime>20110420164828.616</startTime>
    <userTimeZone>1-050000</userTimeZone>
    <localCallId>319:0</localCallId>
    <remoteCallId>315:0</remoteCallId>
    <answerIndicator>Yes</answerIndicator>
    <answerTime>20110420164833.245</answerTime>
    <releaseTime>20110420171439.116</releaseTime>
    <terminationCause>016</terminationCause>
    <callCategory>private</callCategory>
    <chargeIndicator>n</chargeIndicator>
    <releasingParty>remote</releasingParty>
    <userId>north02@txasdev96.net</userId>
    <otherPartyName>john1 north</otherPartyName>

<otherPartyNamePresentationIndicator>Public</otherPartyNamePresentationIndicator
>
    <clidPermitted>Yes</clidPermitted>
    <namePermitted>Yes</namePermitted>
  </basicModule>
  <centrexModule>
    <group>North_as96</group>
    <locationList>
      <locationInformation>
        <location>9726996502@txasdev96.net</location>
        <locationType>Primary Device</locationType>
      </locationInformation>
    </locationList>
    <locationUsage>1565.830</locationUsage>
    <serviceExtensionList>
      <serviceExtension>
        <serviceName>Call Recording</serviceName>
        <recordingTrigger>always-pause-resume</recordingTrigger>
```

```
        <recordingDestination>10.16.150.10:5070</recordingDestination>
        <recordingResult>successful</recordingResult>
      </serviceExtension>
    </serviceExtensionList>
  </centrexModule>
  <ipModule>
    <route>Group</route>
    <codec>PCMA/8000</codec>
    <accessDeviceAddress>10.16.150.2</accessDeviceAddress>
    <accessCallID>BW114828655200411-1111068752@10.16.150.2</accessCallID>
    <codecUsage>1565.821</codecUsage>
  </ipModule>
</cdrData>
```

8 Performance Management

8.1 Performance Counters

The following counters track the activity of the Call Recording service.

Name:	bwCallRecordingAutoRecordings
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	Number of calls that are automatically recorded.
Type:	Counter32
Access:	read/write
Incremented:	Whenever users originate or receive calls with the Call Recording service in *Always* mode.

Name:	bwCallRecordingSPAutoRecordings
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	Number of calls that are automatically recorded for users in the given service provider.
Type:	Counter32
Access:	read/write
Incremented:	Whenever users in the service provider originate or receive calls with the Call Recording service in *Always* mode.

Name:	bwCallRecordingGroupAutoRecordings
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	Number of calls that are automatically recorded for users in the given group.
Type:	Counter32
Access:	read/write
Incremented:	Whenever users in a group originate or receive calls with the Call Recording service in *Always* mode.

Name:	bwCallRecordingOnDemandRecordings
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	Number of calls that are requested to be saved on an on-demand basis.
Type:	Counter32
Access:	read/write
Incremented:	Whenever users request a call to be saved with the Call Recording service in *On-Demand* mode.

Name:	bwCallRecordingSPOnDemandRecordings
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	Number of calls that are requested to be saved on an on-demand basis for users in the given service provider.
Type:	Counter32
Access:	read/write
Incremented:	Whenever users in a service provider request a call to be saved with the Call Recording service in *On-Demand* mode.

Name:	bwCallRecordingGroupOnDemandRecordings
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	Number of calls that are requested to be saved on an on-demand basis for users in the given group.
Type:	Counter32
Access:	read/write
Incremented:	Whenever users in a group request a call to be saved with the Call Recording service in *On-Demand* mode.

Name:	bwCallRecordingOnDemandEnabled
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	Number of calls where on-demand recording is enabled, regardless whether the recordings are triggered to be saved.
Type:	Counter32
Access:	read/write
Incremented:	Whenever users originate or receive calls with the Call Recording service in *On-Demand* mode.

Name:	bwCallRecordingSPOnDemandEnabled
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	Number of calls where on-demand recording is enabled, regardless whether the recordings are triggered to be saved for users in the given service provider.
Type:	Counter32
Access:	read/write
Incremented:	Whenever users in a service provider originate or receive calls with the Call Recording service in *On-Demand* mode.

Name:	bwCallRecordingGroupOnDemandEnabled
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	Number of calls where on-demand recording is enabled, regardless whether the recordings are triggered to be saved for users in the given group.
Type:	Counter32
Access:	read/write
Incremented:	Whenever users in a group originate or receive calls with the Call Recording service in *On-Demand* mode.

8.1.1 New Counters for Start, Stop, Pause, and Resume

Name:	bwCallRecordingPaused
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	Number of times a call recording is paused.
Type:	Counter32
Access:	read/write
Incremented:	Whenever users pause an active call recording.

Name:	bwCallRecordingSPPaused
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	Number of times a call recording is paused for users in the given service provider.
Type:	Counter32
Access:	read/write
Incremented:	Whenever users in the service provider pause an active call recording.

Name:	bwCallRecordingGroupPaused
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	Number of times a call recording is paused for users in the given group.
Type:	Counter32
Access:	read/write
Incremented:	Whenever users in a group pause an active call recording.

Name:	bwCallRecordingResumed
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	Number of times a call recording is resumed.
Type:	Counter32
Access:	read/write
Incremented:	Whenever users resume an active call recording.

Name:	bwCallRecordingSPResumed
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	Number of times a call recording is resumed for users in the given service provider.
Type:	Counter32
Access:	read/write
Incremented:	Whenever users in the service provider resume an active call recording.

Name:	bwCallRecordingGroupResumed
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	Number of times a call recording is resumed for users in the given group.
Type:	Counter32
Access:	read/write
Incremented:	Whenever users in a group resume an active call recording.

8.1.2 New Counters for Video Support

Name:	bwCallRecordingVideoRecordings
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	The number of video calls that are recorded.
Type:	Counter32
Access:	read/write
Incremented:	Whenever a user’s call recording records video.

Name:	bwCallRecordingSPVideoRecordings
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	The number of video calls that are recorded for users in the given service provider.
Type:	Counter32
Access:	read/write
Incremented:	Whenever users in the service provider record video calls.

Name:	bwCallRecordingGroupVideoRecordings
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	The number of video calls that are recorded for users in the given group.
Type:	Counter32
Access:	read/write
Incremented:	Whenever users in a group record video calls.

8.1.3 New Counters for Voice Mail Recording

Name:	bwCallRecordingVoicemailRecording
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	This counter tracks the number of times a voice mail recording starts.
Type:	Counter32
Access:	read/write
Incremented:	This counter is incremented when the Call Recording service starts to record voice messaging for a user.

Name:	bwCallRecordingSPVoicemailRecording
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	This counter tracks the number of times a voice mail recording starts for a user in the service provider.
Type:	Counter32
Access:	read/write
Incremented:	This counter is incremented when the Call Recording service starts to record voice messaging for a user in the service provider.

Name:	bwCallRecordingGroupVoicemailRecording
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	This counter tracks the number of times a voice mail recording starts for a user in the group.
Type:	Counter32
Access:	read/write
Incremented:	This counter is incremented when the Call Recording service starts to record voice messaging for a user in the group.

Name:	bwCallRecordingVoicemailRecordingAbandoned
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	This counter tracks the number of times a voice mail recording is abandoned due to the user accessing their voice portal.
Type:	Counter32
Access:	read/write
Incremented:	This counter is incremented when the Call Recording service is abandoned, when the user accesses their voice portal during the voice message deposit.

Name:	bwCallRecordingSPVoicemailRecordingAbandoned
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	This counter tracks the number of times a voice mail recording is abandoned due to the user in the service provider accessing their voice portal.
Type:	Counter32
Access:	read/write
Incremented:	This counter is incremented when the Call Recording service is abandoned, when the user in the service provider accesses their voice portal during the voice message deposit.

Name:	bwCallRecordingGroupVoicemailRecordingAbandoned
MIB:	BW-Execution.mib
Module:	enterprises.broadsoft.broadworks.executionServer.services.callrecording
Description:	This counter tracks the number of times a voice mail recording is abandoned due to the user in the group accessing their voice portal.
Type:	Counter32
Access:	read/write
Incremented:	This counter is incremented when the Call Recording service is abandoned when the user in the group accesses their voice portal during the voice message deposit.

8.2 Fault Management

The BroadWorks Call Recording service generates notifications and alarms when a fault condition occurs. The following alarm is defined in *alarms* MIB.

The recommendation text for the existing *bwCallRecordingPlatformError* alarm is updated since this alarm is now raised when the call recording fails mid-call due to an INVITE, UPDATE, or INFO not reaching the 3PCR platform or the session audit to the 3PCR platform failing. In addition to its original usage, when the 3PCR platform is unreachable when the recording starts, the alarm clears on the next successful connection to the 3PCR platform. The connection is not attempted until a user attempts to record a call.

Fault Name	Attributes	Value
bwCallRecordingPlatformError	Problem type	Alarm
	Severity range	High (if the 3PCR platform is unreachable). Medium (if the 3PCR platform returns 4xx/5xx/6xx when the Application Server attempts recording).
	Subcomponent	CallP
	Description	The recording attempt was unsuccessful because the 3PCR platform is unreachable (due to connectivity issues) or the 3PCR platform is experiencing issues that are preventing the Application Server from establishing a dialog with it (due to 4xx/5xx/6xx error).
	Problem text	Unable to record on the 3PCR platform at %npRemoteAddress% because it is not responding or it has returned an error.
	Problem text parameters	%npRemoteAddress%: address of the 3PCR platform.

Fault Name	Attributes	Value
	Recommendation text	Check the connectivity to the 3PCR platform. The alarm clears when the system successfully connects to the 3PCR platform.
	Recommendation text parameters	None

9 SIP/MGCP Interface Impacts

9.1 Summary

To support the Call Recording functionality, the following changes to the SIP interface were made to interact with the 3PCR platform. The majority of the changes are based on IETF draft *draft-portman-siprec-protocol03* [1] and *draft-ram-siprec-metadata-format-01* [2]. The changes only affect calls to the 3PCR platform. The following changes were made to support interactions with the 3PCR platform:

- Modifies the following SIP headers:
  - Contact – New feature tags
  - Content-Type – New mime type
  - Content-Disposition – New disposition type
- Adds two new SIP headers:
  - Info-package
    - Adds an *x-broadworks-callrecording* INFO event package
  - Recv-Info
- Adds a new SDP a-line *a=label:* to identify recording streams
- Adds a new XML message body:
  - *rs-metadata+xml*
- Adds Cisco BroadWorks-specific XML extension data:
  - *broadworks-recording-metadata*
- Sends INFO message with info-package package, *x-broadworks-callrecording*

9.1.1 Call Center and Route Point

To enhance the Call Recording functionality, new metadata elements have been added to the Cisco BroadWorks-specific extension data in the interface to the 3PCR platform. This data conveys information specific for agent recordings involving a call center or route point.

The new metadata captures the following:

- The user ID of the call center/route point being used to reach the agent.
- The name configured against the call center/route point.
- The address of the call center/route point being used to reach the agent (for example, the primary DN or a DNIS configured against the call center/route point).

The new metadata impacts the *broadworks-recording-metadata* definition of the 3PCR platform interface.

For interoperability, these new elements are only included in the metadata sent to the 3PCR platforms that support *broadworks-recording-metadata* schema version 2.0 or later. The administrator can indicate the schema version against each 3PCR platform configuration.

9.1.2 Voice Mail Recording

To report to the 3PCR platform that this is a voice message recording, the Cisco BroadWorks-specific extension data contains new metadata element. The recording type in the Cisco BroadWorks extension now allows the new "voicemail" type.

9.1.3 Controls for IP Phones

To provide recording control interactions with the recording-aware UA, the following changes were made to the SIP interface:

- Added a new event to the *as-feature-event* event package: CallRecordingModeEvent
- Added a new option tag to the *Supported* header: *record-aware*
- Added a new optio*n tag to* the *Require* header: *recor*d-aware
- Added two new SDP attributes:
  - *record*
  - *recordpref*

9.1.4 Visual Security Classification for Active Calls

The Cisco BroadWorks-specific extension metadata contains a new metadata element, *callClassification*, which denotes the highest security classification level reached for call Adds.

9.1.5 draft-portman-siprec-protocol-03

This section highlights the portions of the protocol draft that the Cisco BroadWorks Application Server sends or expects to receive from the 3PCR platform. The Cisco BroadWorks Application Server does not support the following sections of the *draft-portman-siprec-protocol-03*:

- section 6: SIP extensions for recording-aware devices
- the new mime type application/rs-metadata-request in section 7.2.2
- the new Info-Package recording-session-srs

For more information, see section *4.2.3 draft-portman-siprec-protocol-03.*

9.1.6 draft-ram-siprec-metadata-format-01

The metadata format is followed in its entirety. This is reflected in the XML schema defined in section *9.6.1 rs-metadata+xml*.

This feature also defines Cisco BroadWorks-specific extension data to be included as part of the extension data for the communication session. For more information on the extension data, see sections *4.3 Cisco BroadWorks-specific Extension Data* and *9.6.2 broadWorksRecordingMetadata*.

9.2 SIP Headers

9.2.1 Contact

Two new contact parameters are added to the *Contact* header. The new parameters are:

- *src*
- *srs*

The new parameters are used to specify whether the message is from a Session Recording Client (*src*) or a Session Recording Server (*srs*). These two parameters are added for compliance with *draft-portman-siprec-protocol-03* [1]. The Cisco BroadWorks Application Server sends *src* in the contact to the 3PCR server and may receive the *srs* parameter back from the 3PCR server.

Only the relevant parts of the *Contact* header Augmented Backus-Naur Form (ABNF) are shown in the following example. The others are unchanged from *RFC 3261* [3].

```
Contact          =  ("Contact" | "m" ) HCOLON
                    ( STAR | (contact-param *(COMMA contact-param)))
contact-param  =  (name-addr | addr-spec) *(SEMI contact-params)
contact-params     =  c-p-q | c-p-expires | "src" | "srs"
                        | contact-extension
contact-extension  =  generic-param
```

9.2.2 Content-Type

A new mime type is added to the list of supported mime types. The new mime type is used to identify the recording session metadata. This metadata is used by the recording server to store the recording for later retrieval. This mime type is registered with Internet Assigned Numbers Authority (IANA) in *draft-portman-siprec-portocol-03* [1].

Only the relevant parts of the *Content-Type* header ABNF are shown in the following example. The others are unchanged from *RFC 3261* [3].

```
Content-Type      =  ( "Content-Type" | "c" ) HCOLON media-type
media-type        =  m-type SLASH m-subtype *(SEMI m-parameter)
m-subtype         =  extension-token | iana-token | "rs-metadata+xml"
```

9.2.3 Content-Disposition

A new content-disposition type is added to identify this data as belonging to a recording session. This new disposition type is registered with IANA by *draft-portman-siprec-portocol-03* [1].

Only the relevant parts of the *Content-Disposition* header ABNF are shown in the following example. The others are unchanged from *RFC 3261* [3].

```
Content-Disposition    =  "Content-Disposition" HCOLON
                          disp-type *( SEMI disp-param )
disp-type           =  "render" | "session" | "icon" | "alert" |
                       "recording-session" | disp-extension-token
```

9.2.4 Info-Package

The *Info-Package* header is defined in *RFC 6086* [5]. It is used to specify an INFO event package that is included in an INFO message. This feature sends the *x-broadworks-callrecording* package to indicate that the 3PCR platform should save the recording that belongs to the SIP dialog to which this INFO message belongs.

The format of the *Info-Package* header is shown in the following example.

```
Info-Package = "Info-Package" HCOLON Info-package-type
Info-package-type = Info-package-name *( SEMI Info-package-param)
Info-package-name = token | "x-broadworks-callrecording"
Info-package-param = generic-param
```

9.2.5 Recv-Info

The *Recv-Info* header is defined in *RFC 6086* [5]. It is used to specify what information event packages a UA can support. This feature includes an *empty Recv-Info* header in the initial INVITE to inform the 3PCR platform that the INFO event package is supported. This feature expects that the Call Recording platform should include a *Recv-Info* header with the *x-broadworks-callrecording* event package specified to let the Application Server know that it supports the *On-Demand* recording. The *Recv-Info* should be sent by the 3PCR platform in the *200 OK* response to the initial INVITE for the Application Server to know that it can send the INFO message.

The format of the Recv-Info header is shown in the following example.

```
Recv-Info = "Recv-Info" HCOLON [Info-package-list]
Info-package-list = Info-Package-type * ( COMA Info-package-type )
Info-package-type = Info-package-name * ( SEMI Info-package-param )
Info-package-name = token | "x-broadworks-callrecording"
Info-package-param = generic-param
```

9.3 SIP Method

9.3.1 INFO

The INFO method is defined in *RFC 6086* [5]. In this feature, there is a need to inform the 3PCR server that a call recording needs to be kept. This is required when the call recording mode for the user is *On-Demand*. The INFO method is used to inform the 3PCR platform to keep the call recording. The INFO message must be sent during the lifetime of the call. If the 3PCR platform does not receive an INFO message prior to the recording session ending, the recording is deleted.

To identify the call recording to the 3PCR platform, the following SIP headers are set as follows. All other headers not mentioned are set according to the description in *RFC 6086*.

Header	VALUE	Description/reason
Info-Package	"x-broadworks-callrecording"	This specifies that this is for the Call Recording Server to keep the currently active recording.

9.4 Call Control for IP Phones Events and Headers

9.4.1 CallRecordingModeEvent

The *CallRecordingModeEvent* is sent by the Cisco BroadWorks Application Server to the recording-aware UA in the body of a NOTIFY upon initial subscription to the Device Feature Synchronization event package (*as-feature-event*) and upon modification of the provisioned recording mode for the call recording user.

The value in the *CallRecordingModeEvent* informs the recording-aware UA as to which recording mode is currently provisioned against the call recording user. The recording mode values can be "always", "always-pause-resume", "on-demand", "on-demand-user-start", or "never".

If the Call Recording service is unassigned from the user, the value of the recording mode is empty.

The following schema describes the new *CallRecordingModeEvent*.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
xmlns:xBw="http://schema.broadsoft.com/as-feature-event/XML"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation>Call Recording Mode Event</xs:documentation>
  </xs:annotation>
  <xs:element name="CallRecordingModeEvent">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="callRecordingMode" type="xBw:RecordingMode"
nillable="true">
            <xs:annotation>
              <xs:documentation>
                The provisioned call recording mode for the user.
                The element is nil if the user does not have the service.
              </xs:documentation>
            </xs:annotation>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:simpleType name="RecordingMode">
    <xs:annotation>
      <xs:documentation>
        The recording mode of the Call Recording user:
        ALWAYS indicates calls originated by or received by the user are
always recorded and saved.
        ALWAYS-PAUSE-RESUME indicates that calls originated or received
by the user are always recorded and saved. The user also has the ability
to pause and resume recordings.
        ON-DEMAND indicates that calls can be selectively recorded and
saved on an on-demand basis. The user also has the ability to pause and
resume recordings.
        ON-DEMAND-USER-START indicates that the call recording does not
start until initiated by the user. The user also has the ability to
pause, resume and stop recordings.
        NEVER indicates calls are never recorded.
      </xs:documentation>
    </xs:annotation>
    <xs:restriction base="NonEmptyToken">
      <xs:enumeration value="always" />
      <xs:enumeration value="always-pause-resume"/>
      <xs:enumeration value="on-demand" />
      <xs:enumeration value="on-demand-user-start"/>
      <xs:enumeration value="never" />
    </xs:restriction>
  </xs:simpleType>
</xs:schema>
```

9.4.2 Supported Header

A new option tag is added to the *Supported* header. This tag is sent by the recording-aware UA, in addition to the Cisco BroadWorks Application Server, to specify that the new recording indicator attributes in the SDP are supported. The new option tag is "record-aware".

The new option tag is registered in IANA by *draft-ietf-siprec-protocol-09* [19].

Only the relevant parts of the *Supported* header ABNF are shown in the following example. The others are unchanged from *RFC 3261* [3].

```
Supported  =  ( "Supported" / "k" ) HCOLON

              [option-tag *(COMMA option-tag)]
option-tag     =  token | "100rel" | "pref" | "timer" | "event-list" |
                  "early-session" | "broadworkscalltypequery" | "gin" |
                  "altc" | "record-aware"
```

9.4.3 Require Header

A new option tag is added to the *Require* header. This tag may be sent by the recording-aware UA to specify that they require the new recording indicator attributes in the SDP, added by this feature. The new tag is "record-aware".

The new option tag is registered in IANA by *draft-ietf-siprec-protocol-09,* section *11.1.2* [19].

Only the relevant parts of the *Require* header ABNF are shown in the following example. The others are unchanged from *RFC 3261* [3].

```
Require  =  ( "Require" / "k" ) HCOLON
              [option-tag *(COMMA option-tag)]
option-tag     =  token | "100rel" | "pref" | "timer" | "event-list" |
                  "early-session" | "broadworkscalltypequery" | "gin" |
                  "altc" | "record-aware"
```

9.5 SDP Body

When two RTP streams are sent to the 3PCR platform, each stream represents the audio from a different participant in the call being recorded. For the 3PCR platform to coordinate the streams with the metadata it receives, each media description is labeled. To do this, the *label* attribute in the SDP is used. The *label* attribute is defined in *RFC 4574* [4]. The following example shows two RTP audio streams labeled 1 and 2.

```
v=0
o=SRC 0 0 IN IP4 172.22.3.8
s=SRC
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 12243 RTP/AVP 0 4 8
a=sendonly
a=label: 2
```

9.5.1 Call Control IP Phones

Two new SDP attributes are supported by the Cisco BroadWorks Application Server:

- *recordpref*
- *record*

9.5.1.1 recordpref

The *recordpref* attribute is sent by the Cisco BroadWorks recording-aware UA to indicate how it prefers the Cisco BroadWorks Application Server to handle the recording of the call. This attribute is defined in *draft-ietf-siprec-protocol-09,* section *7.3.2* [19].

The following shows the ABNF of the *recordpref* attribute.

```
attribute /= recordpref-attr

; attribute defined in RFC 4566

recordpref-attr = "a=recordpref:" pref

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

- "on" – Sets the preference to record if it has not already been started. If the recording is currently paused, the preference is to resume recording.
- "off" – Sets the preference for no recording. If recording has already been started, then the preference is to stop the recording.
  - "pause" – Sets the preference to pause the recording if the recording is currently in progress.
  - "nopreference" – Indicates that the UA has no preference on recording.

9.5.1.2 record

The *record* attribute is sent by the Cisco BroadWorks Application Server to report the status of the call recording stream to the Cisco BroadWorks recording-aware UA, on behalf of the call being recorded. This attribute is defined in *draft-ietf-siprec-protocol-09,* section *7.1.2* [19].

The following shows the ABNF of the *record* attribute.

```
attribute /= record-attr

; attribute defined in RFC 4566

record-attr = "record:" indication

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

- "on" – Recording is in progress.
- "off" – No recording is in progress.
- "paused" – Recording is in progress but media is paused.

9.6 SIP Message Body

9.6.1 rs-metadata+xml

This message body is defined in *draft-ram-siprec-metadata-format-01* [2]. This data defines the recording session. The Cisco BroadWorks-specific information is included in the extension data for the communication session element. For information on the XML schema for the Cisco BroadWorks-specific information, see section *9.6.2 broadWorksRecordingMetadata*.

The 3PCR platforms should store and index the recordings based on the Cisco BroadWorks-specific extension. This version deviates from the XML Schema Definition (XSD) documented in the draft in one area. The *streamDirection* complex type is modified to match the examples in the draft. Its internal tag is changed to "id" from "label", and the type is changed from "xs:string" to "xs:urnuuid".

Following is an example of the XML schema information for this body format.

```
<?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-metadata"
    type="tns:recording-metadata" maxOccurs="unbounded"/>
<xs:complexType name="recording-metadata">
  <xs:sequence>
    <xs:element name="datamode" type="xs:dataMode" minOccurs="0"/>
    <xs:element name="recording" type="tns:recording"
      minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="group" type="tns:group"
      minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="session" type="tns:session"
      minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="participant" type="tns:participant"
      minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="stream" type="tns:stream"
      minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="extensiondata" type="tns:extensiondata"
      minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="recording">
  <xs:sequence>
    <xs:element name="requestor" type="xs:requestor" minOccurs="0"/>
    <xs:element name="type" type="xs:type" minOccurs="0"/>
    <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="xs:urnuuid" use"required"/>
</xs:complexType>
<xs:complexType name="group">
  <xs:sequence>
    <xs:element name="initiator" type="xs:anyURI" 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:sequence>
  <xs:attribute name="id" type="xs:urnuuid" use"required"/>
  <xs:attribute name="recording" type="xs:urnuuid" use"required"/>
</xs:complexType>
<xs:complexType name="session">
  <xs:sequence>
    <xs:element name="start-time" type="xs:dateTime" minOccurs="0"/>
    <xs:element name="stop-time" type="xs:dateTime" minOccurs="0"/>
    <xs:element name="reason" type="xs:string" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:urnuuid" use"required"/>
```

```
  <xs:attribute name="group" type="xs:urnuuid" use"required"/>
</xs:complexType>
<xs:complexType name="participant">
  <xs:sequence>
    <xs:element name="aor" type="xs:anyURI" maxOccurs="1"/>
    <xs:element name="send" type="xs:streamDirection" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="recv" type="xs:streamDirection" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="extensiondata" type="tns:extensiondata"
      minOccurs="0"/>
    <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="xs:urnuuid" use"required"/>
  <xs:attribute name="session" type="xs: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="xs: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:sequence>
  <xs:attribute name="id" type="xs:urnuuid" use"required"/>
  <xs:attribute name="session" type="xs:urnuuid" use"required"/>
</xs:complexType>
<xs:simpleType name="streamMode">
  <xs:restriction base="xs:string">
    <xs:pattern value="mixed|separate"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="requestor">
  <xs:restriction base="xs:string">
    <xs:pattern value="SRC|SRS"/>
  </xs:restriction>
</xs:simpleType>
<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 name="type">
  <xs:restriction base="xs:string">
    <xs:pattern value="selective|persistant"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="dataMode">
  <xs:restriction base="xs:string">
    <xs:pattern value="Complete|partial"/>
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="extensiondata">
  <xs:sequence>
    <xs:element name="string" type="xs:any" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:urnuuid" use"required"/>
  <xs:attribute name="parent" type="xs:urnuuid" use"required"/>
</xs:complexType>
```

```
<xs:complexType name="streamDirection">
  <xs:sequence>
    <xs:element name="id" type="xs:urnuuid" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
</xs:schema>
```

9.6.2 broadWorksRecordingMetadata

This message body is used in two places. The first is in the call setup. In this instance, this data contains the information from the Cisco BroadWorks Application Server that identifies the call. This information should be used by the 3PCR platform to index and store the call recording. The second usage of this data is in the re-INVITE or the UPDATE messages when the metadata is being updated due to a change in the call being recorded. The primary cases for the change are if the call is transferred.

Following is an example of the XML schema for this message body.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
targetNamespace="http://schema.broadsoft.com/broadworksCallRecording"
  xmlns:tns="http://schema.broadsoft.com/broadworksCallRecording"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified"
  version="1.0">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
  schemaLocation="http://www.w3.org/2001/xml.xsd"/>
  <xs:element name="BroadWorks-Recording-metadata"
  type="tns:BroadWorks-recording-metadata"/>
  <xs:complexType name="broadWorksRecordingMetadata">
    <xs:sequence>
      <xs:element name="extTrackingID" type="xs:string"
        minOccurs="1"/>
      <xs:element name="serviceProviderID" type="xs:string"
        minOccurs="0"/>
      <xs:element name="groupID" type="xs:string" minOccurs="0"/>
      <xs:element name="userID" type="xs:string" minOccurs="0"/>
      <xs:element name="callID" type="xs:string" minOccurs="0"/>
      <xs:element name="callType" type="tns:callTypeInfo" minOccurs="0"/>
      <xs:element name="recordingType" type="tns:recordingType"
        minOccurs="0"/>
      <xs:element name="redirectedCall" type="tns:redirectedCall"
        minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="callTypeInfo">
    <xs:choice>
      <xs:element name="origCall" type="tns:origCallDetails"/>
      <xs:element name="termCall" type="tns:termCallDetails"/>
    </xs:choice>
  </xs:complexType>
  <xs:complexType name="origCallDetails">
    <xs:sequence>
      <xs:element name="callingPartyNumber" type="xs:string" />
      <xs:element name="calledPartyNumber" type="xs:string"/>
      <xs:element name="dialedDigits" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="termCallDetails">
```

```
    <xs:sequence>
      <xs:element name="callingPartyNumber" type="xs:string"/>
      <xs:element name="calledPartyNumber" type="xs:string"/>
      <xs:element name="redirectInfo" type="tns:redirectInfo"
        minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="redirectInfo">
    <xs:sequence>
      <xs:element name="lastRedirectNumber" type="xs:string"
        minOccurs="0"/>
      <xs:element name="origCalledNumber" type="xs:string"
        minOccurs="0"/>
      <xs:element name="numOfRedirections" type="xs:integer"
        minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="redirectedCall">
    <xs:sequence>
      <xs:element name="newExtTrackingID" type="xs:string"
        minOccurs="0"/>
      <xs:element name="redirectedFromPartyNumber" type="xs:string"/>
      <xs:element name="redirectedToPartyNumber" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="recordingType">
    <xs:restriction base="xs:string">
      <xs:pattern value="on|demand"/>
    </xs:restriction>
  </xs:simpleType>
</xs:schema>
```

9.6.2.1 Version 2.0

This message body is used in two places. The first usage is in the call setup to the 3PCR platform. In this instance, the data contains information from the Cisco BroadWorks Application Server, which identifies the call. This information should be used by the 3PCR platform to index and store the call recording. The second usage of this data is in the re-INVITE or the UPDATE message when the metadata is being updated due to a change in the call being recorded.

The following schema definition shows the new and changed elements highlighted in **bold** font. For the original schema definition, see the *Call Recording Interface Feature Description*.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
targetNamespace="http://schema.broadsoft.com/broadworksCallRecording"
  xmlns:tns="http://schema.broadsoft.com/broadworksCallRecording"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified"
  version="2.0">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
  schemaLocation="http://www.w3.org/2001/xml.xsd"/>
  <xs:element name="BroadWorks-Recording-metadata"
  type="tns:BroadWorks-recording-metadata"/>
  <xs:complexType name="broadWorksRecordingMetadata">
    <xs:sequence>
      <xs:element name="extTrackingID" type="xs:string"
```

```
      minOccurs="1"/>
    <xs:element name="serviceProviderID" type="xs:string"
      minOccurs="0"/>
    <xs:element name="groupID" type="xs:string" minOccurs="0"/>
    <xs:element name="userID" type="xs:string" minOccurs="0"/>
    <xs:element name="callID" type="xs:string" minOccurs="0"/>
    <xs:element name="callType" type="tns:callTypeInfo" minOccurs="0"/>
    <xs:element name="recordingType" type="tns:recordingType"
      minOccurs="0"/>
    <xs:element name="redirectedCall" type="tns:redirectedCall"
      minOccurs="0"/>
    <xs:element name="acd" type="tns:acdDetails" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="callTypeInfo">
  <xs:choice>
    <xs:element name="origCall" type="tns:origCallDetails"/>
    <xs:element name="termCall" type="tns:termCallDetails"/>
  </xs:choice>
</xs:complexType>
<xs:complexType name="origCallDetails">
  <xs:sequence>
    <xs:element name="callingPartyNumber" type="xs:string" />
    <xs:element name="calledPartyNumber" type="xs:string"/>
    <xs:element name="dialedDigits" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="termCallDetails">
  <xs:sequence>
    <xs:element name="callingPartyNumber" type="xs:string"/>
    <xs:element name="calledPartyNumber" type="xs:string"/>
    <xs:element name="redirectInfo" type="tns:redirectInfo"
      minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="redirectInfo">
  <xs:sequence>
    <xs:element name="lastRedirectNumber" type="xs:string"
      minOccurs="0"/>
    <xs:element name="origCalledNumber" type="xs:string"
      minOccurs="0"/>
    <xs:element name="numOfRedirections" type="xs:integer"
      minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="redirectedCall">
  <xs:sequence>
    <xs:element name="newExtTrackingID" type="xs:string"
      minOccurs="0"/>
    <xs:element name="redirectedFromPartyNumber" type="xs:string"/>
    <xs:element name="redirectedToPartyNumber" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
<xs:simpleType name="recordingType">
  <xs:restriction base="xs:string">
    <xs:pattern value="on|demand"/>
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="acdDetails">
  <xs:sequence>
    <xs:element name="acdUserId" type="xs:string" />
    <xs:element name="acdName" type="xs:string"/>
```

```
      <xs:element name="acdNumber" type="tns:acdNumberDetails"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="acdNumberDetails">
    <xs:simpleContent>
      <xs:extension base="NonEmptyURI">
        <xs:attribute name="countryCode" type="xs:string"
use="optional"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:simpleType name="NonEmptyURI">
    <xs:restriction base="xs:anyURI">
      <xs:minLength value="1"/>
    </xs:restriction>
  </xs:simpleType>
</xs:schema>
```

9.6.2.2 Version 3.0

The following schema definition shows the new and changed elements highlighted in **bold** font.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
targetNamespace="http://schema.broadsoft.com/broadworksCallRecording"
  xmlns:tns="http://schema.broadsoft.com/broadworksCallRecording"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified"
  version="3.0">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
  schemaLocation="http://www.w3.org/2001/xml.xsd"/>
  <xs:element name="BroadWorks-Recording-metadata"
  type="tns:BroadWorks-recording-metadata"/>
  <xs:complexType name="broadWorksRecordingMetadata">
    <xs:sequence>
      <xs:element name="extTrackingID" type="xs:string"
        minOccurs="1"/>
      <xs:element name="serviceProviderID" type="xs:string"
        minOccurs="0"/>
      <xs:element name="groupID" type="xs:string" minOccurs="0"/>
      <xs:element name="userID" type="xs:string" minOccurs="0"/>
      <xs:element name="callID" type="xs:string" minOccurs="0"/>
      <xs:element name="callType" type="tns:callTypeInfo" minOccurs="0"/>
      <xs:element name="recordingType" type="tns:recordingType"
        minOccurs="0"/>
      <xs:element name="redirectedCall" type="tns:redirectedCall"
        minOccurs="0"/>
      <xs:element name="acd" type="tns:acdDetails" minOccurs="0"/>
     <xs:element name="callClassification" type="xs:string"
minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="callTypeInfo">
    <xs:choice>
      <xs:element name="origCall" type="tns:origCallDetails"/>
      <xs:element name="termCall" type="tns:termCallDetails"/>
    </xs:choice>
  </xs:complexType>
```

```
  <xs:complexType name="origCallDetails">
    <xs:sequence>
      <xs:element name="callingPartyNumber" type="xs:string" />
      <xs:element name="calledPartyNumber" type="xs:string"/>
      <xs:element name="dialedDigits" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="termCallDetails">
    <xs:sequence>
      <xs:element name="callingPartyNumber" type="xs:string"/>
      <xs:element name="calledPartyNumber" type="xs:string"/>
      <xs:element name="redirectInfo" type="tns:redirectInfo"
        minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="redirectInfo">
    <xs:sequence>
      <xs:element name="lastRedirectNumber" type="xs:string"
        minOccurs="0"/>
      <xs:element name="origCalledNumber" type="xs:string"
        minOccurs="0"/>
      <xs:element name="numOfRedirections" type="xs:integer"
        minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="redirectedCall">
    <xs:sequence>
      <xs:element name=”newExtTrackingID” type=”xs:string”
        minOccurs=”0”/>
      <xs:element name="redirectedFromPartyNumber" type="xs:string"/>
      <xs:element name="redirectedToPartyNumber" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="recordingType">
    <xs:restriction base="xs:string">
      <xs:pattern value="on|demand|voicemail"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="acdDetails">
    <xs:sequence>
      <xs:element name="acdUserId" type="xs:string" />
      <xs:element name="acdName" type="xs:string"/>
      <xs:element name="acdNumber" type="tns:acdNumberDetails"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="acdNumberDetails">
    <xs:simpleContent>
      <xs:extension base="NonEmptyURI">
        <xs:attribute name="countryCode" type="xs:string"
use="optional"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:simpleType name="NonEmptyURI">
    <xs:restriction base="xs:anyURI">
      <xs:minLength value="1"/>
    </xs:restriction>
  </xs:simpleType>
</xs:schema>
```

9.7 Message Examples

9.7.1 INVITE to 3PCR for Call Center and Route Point

This message shows an INVITE to a 3PCR platform when a call center agent has the Call Recording service enabled. This example shows how the *broadworks-recording-metadata* appears based on the schema definition *broadWorksRecordingMetadata (Version 2.0)*. The new elements are highlighted in **bold** font in this example.

```
INVITE sip:3PCR@recorder.broadsoft.com:5060;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26f33f26931B5DED
From: <sip:as1.broadsoft.com>;tag=B6CB9EB1-8AE7B75C
To: sip:3PCR@recorder.broadsoft.com
Call-ID:BW153311499010488569905@10.16.134.17
CSeq:25 INVITE
Contact:<sip:as1.broadsoft.com:5060>;src
Supported:100rel
Recv-Info:
Allow:ACK,BYE,CANCEL,INFO,INVITE,OPTIONS,PRACK,REFER,NOTIFY,UPDATE
Accept:application/media_control+xml,application/sdp,application/x-broadworks-call-
center+xml
Max-Forwards:10
Content-Type:multipart/mixed;boundary=UniqueBroadWorksBoundary
Content-Length: ...


--UniqueBroadWorksBoundary
Content-Type:application/rs-metadata+xml
Content-Disposition:recording-session
Content-Length:2224
<?xml version="1.0" encoding="UTF-8"?>
<recording_metadata xmlns="urn:ietf:params:xml:ns:siprec">
  <dataMode>complete</dataMode>
  <recording id="urn:uuid:b1981e40-e2fc-49ec-b767-
9de4caa0174c"><requestor>SRC</requestor><type>selective</type></recording>
  <group id="urn:uuid:c52a924c-5500-464f-88e0-a3075f1d7602"
recording="urn:uuid:b1981e40-e2fc-49ec-b767-9de4caa0174c">
    <initiator>sip:agent@txasdev96.net</initiator>
  </group>
  <session id="urn:uuid:44ab6658-2997-4490-aa19-898fac300e45"
group="urn:uuid:c52a924c-5500-464f-88e0-a3075f1d7602">
    <start-time>2012-05-22T17:58:43-0500</start-time>
  </session>
  <participant id="urn:uuid:cfecac9c-dc21-46c5-92f8-350c47d7f07c"
session="urn:uuid:44ab6658-2997-4490-aa19-898fac300e45">
    <aor>sip:agent@txasdev96.net</aor>
    <send>
      <id>urn:uuid:ed743db7-b4fb-41ca-8b45-253ed1002aaf</id>
    </send>
  </participant>
  <participant id="urn:uuid:18a31609-f61f-4bb2-b3c4-27035851bf2e"
session="urn:uuid:44ab6658-2997-4490-aa19-898fac300e45">
    <aor>sip:500@10.16.134.101</aor>
      <send>
        <id>urn:uuid:2bf120cb-fd84-4c7c-af8f-9cc13217659a</id>
      </send>
  </participant>
  <stream id="urn:uuid:ed743db7-b4fb-41ca-8b45-253ed1002aaf"
session="urn:uuid:44ab6658-2997-4490-aa19-898fac300e45">
    <label>1</label>
    <mode>separate</mode>
  </stream>
  <stream id="urn:uuid:2bf120cb-fd84-4c7c-af8f-9cc13217659a"
session="urn:uuid:44ab6658-2997-4490-aa19-898fac300e45">
    <label>2</label>
    <mode>separate</mode>
  </stream>
  <extensiondata id="urn:uuid:0862970e-93d9-4bcf-a827-45c3d9cfe1ec"
parent="urn:uuid:44ab6658-2997-4490-aa19-898fac300e45">
```

```
    <broadWorksRecordingMetadata
xmlns=http://schema.broadsoft.com/broadworksCallRecording version="2.0">
      <extTrackingID>5:1</extTrackingID>
      <serviceProviderID>TxASDev</serviceProviderID>
      <groupID>North_as96</groupID>
      <userID>agent@txasdev96.net</userID>
      <callID>callhalf-135:0</callID>
      <callType>
        <termCall>
          <callingPartyNumber>sip:500@10.16.134.101</callingPartyNumber>
          <calledPartyNumber>sip:502@10.16.134.101</calledPartyNumber>
          <redirectInfo>
            <lastRedirectNumber>sip:502@10.16.134.101</lastRedirectNumber>
            <origCalledNumber>sip:+19726996519@10.16.134.101</origCalledNumber>
            <numOfRedirections>1</numOfRedirections>
          </redirectInfo>
        </termCall>
      </callType>
      <recordingType>on</recordingType>
      <acd>
        <acdUserId>BroadSoftCallCenter@txasdev96.net</acdUserId>
        <acdName>BroadSoft Call Center</acdName>
        <acdNumber countryCode="1">tel:+19726996519</acdNumber>
      </acd>
    </broadWorksRecordingMetadata>
  </extensiondata>
</recording_metadata>
--UniqueBroadWorksBoundary
Content-Type:application/sdp
Content-Length:244
v=0
o=BroadWorks 102 1 IN IP4 10.16.120.22
s=-
c=IN IP4 10.16.120.22
t=0 0
m=audio 11490 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=ptime:20
a=sendonly
a=label:1
m=audio 11494 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=ptime:20
a=sendonly
a=label:2
-UniqueBroadWorksBoundary
```

9.7.2 INVITE to 3PCR for Voice Mail Recording

This message shows an invitation to a 3PCR platform. It shows examples of the changes to the SIP messages defined in sections *9.2.1 Contact*, *9.2.2 Content-Type*, *9.2.3 Content-Disposition*, *9.5 SDP Body*, *9.2.5 Recv-Info*, *9.6.1 rs-metadata+xml*, and *9.6.2 broadWorksRecordingMetadata*.

NOTE: The recording type now has a value of "voicemail" in Version 3.0.

```
INVITE sip:3PCR@recorder.broadsoft.com:5060;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26f33f26931B5DED
From: <sip:as1.broadsoft.com>;tag=B6CB9EB1-8AE7B75C
To: sip:3PCR@recorder.broadsoft.com
Call-ID:BW153311499010488569905@10.16.134.17
CSeq:25 INVITE
Contact:<sip:as1.broadsoft.com:5060>;src
Supported:100rel
```

```
Recv-Info:
Allow:ACK,BYE,CANCEL,INFO,INVITE,OPTIONS,PRACK,REFER,NOTIFY,UPDATE
Accept:application/media_control+xml,application/sdp,application/x-broadworks-call-
center+xml
Max-Forwards:10
Content-Type:multipart/mixed;boundary=UniqueBroadWorksBoundary
Content-Length: ...


--UniqueBroadWorksBoundary
Content-Type:application/rs-metadata+xml
Content-Disposition:recording-session
Content-Length:2224
<?xml version="3.0" encoding="UTF-8"?>
<recording_metadata xmlns="urn:ietf:params:xml:ns:siprec">
  <dataMode>complete</dataMode>
  <recording id="urn:uuid:b1981e40-e2fc-49ec-b767-
9de4caa0174c"><requestor>SRC</requestor><type>selective</type></recording>
  <group id="urn:uuid:c52a924c-5500-464f-88e0-a3075f1d7602"
recording="urn:uuid:b1981e40-e2fc-49ec-b767-9de4caa0174c">
    <initiator>sip:north00@broadsoft.com</initiator>
  </group>
  <session id="urn:uuid:44ab6658-2997-4490-aa19-898fac300e45"
group="urn:uuid:c52a924c-5500-464f-88e0-a3075f1d7602">
    <start-time>2012-05-22T17:58:43-0500</start-time>
  </session>
  <participant id="urn:uuid:cfecac9c-dc21-46c5-92f8-350c47d7f07c"
session="urn:uuid:44ab6658-2997-4490-aa19-898fac300e45">
    <aor>sip:north00@broadsoft.com</aor>
    <send>
      <id>urn:uuid:ed743db7-b4fb-41ca-8b45-253ed1002aaf</id>
    </send>
  </participant>
  <participant id="urn:uuid:18a31609-f61f-4bb2-b3c4-27035851bf2e"
session="urn:uuid:44ab6658-2997-4490-aa19-898fac300e45">
    <aor>sip:13019779440@broadsoft.com</aor>
      <send>
        <id>urn:uuid:2bf120cb-fd84-4c7c-af8f-9cc13217659a</id>
      </send>
  </participant>
  <stream id="urn:uuid:ed743db7-b4fb-41ca-8b45-253ed1002aaf"
session="urn:uuid:44ab6658-2997-4490-aa19-898fac300e45">
    <label>1</label>
    <mode>separate</mode>
  </stream>
  <stream id="urn:uuid:2bf120cb-fd84-4c7c-af8f-9cc13217659a"
session="urn:uuid:44ab6658-2997-4490-aa19-898fac300e45">
    <label>2</label>
    <mode>separate</mode>
  </stream>
  <extensiondata id="urn:uuid:0862970e-93d9-4bcf-a827-45c3d9cfe1ec"
parent="urn:uuid:44ab6658-2997-4490-aa19-898fac300e45">
    <broadWorksRecordingMetadata
xmlns=http://schema.broadsoft.com/broadworksCallRecording version="2.0">
      <extTrackingID>5:1</extTrackingID>
      <serviceProviderID>TxASDev</serviceProviderID>
      <groupID>North_as96</groupID>
      <userID>north00@broadsoft.com</userID>
      <callID>callhalf-135:0</callID>
      <callType>
        <termCall>
          <callingPartyNumber>sip:13019779440@broadsoft.com</callingPartyNumber>
          <calledPartyNumber>sip:9725551212@broadsoft.com</calledPartyNumber>
        </termCall>
      </callType>
      <recordingType>voicemail</recordingType>
    </broadWorksRecordingMetadata>
  </extensiondata>
</recording_metadata>
--UniqueBroadWorksBoundary
```

```
Content-Type:application/sdp
Content-Length:244
v=0
o=BroadWorks 102 1 IN IP4 10.16.120.22
s=-
c=IN IP4 10.16.120.22
t=0 0
m=audio 11490 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=ptime:20
a=sendonly
a=label:1
m=audio 11494 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=ptime:20
a=sendonly
a=label:2
--UniqueBroadWorksBoundary—
```

9.7.3 INFO Example

The following INFO method is sent to trigger the 3PCR platform to permanently store the call recording setup by the INVITE in the previous example. The INFO message does not contain a message body, since the call recording to be kept is the recording that was established when this SIP dialog was started. It gives examples of the SIP changes from section 9.2.4 Info-Package.

```
INFO sip:3PCR@recorder.broadsoft.com:5060;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26ff4f2625424521
From: <sip:as1.broadsoft.com>;tag=B6CB9EB1-8AE7B5DC
To: sip:3PCR@recorder.broadsoft.com;tag=3134134f
Call-ID:BW153311499010411112159050@10.16.134.17
CSeq:157450408 INFO
Max-Forwards: 70
Info-Package: x-broadworks-callrecording
Contact: <sip:as1.broadsoft.com>
Content-Length: 0
```

9.7.4 UPDATE Example

The UPDATE message is used to update the metadata in case of changes. The following message example shows an update when the subscriber transfers the call. The only change is to add the redirecting call information to the metadata. This is the update prior to the transfer being processed. It provides examples of the changes in sections 9.2.1 Contact, 9.2.2 Content-Type, 9.2.3 Content-Disposition, 9.5 SDP Body, and 9.6 SIP Message Body.

```
UPDATE sip:3PCR@recorder.broadsoft.com:5060;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26f33f26931B5DED
From: <sip:as1.broadsoft.com>;tag=B6CB9EB1-8AE7B75C
To: sip:3PCR@recorder.broadsoft.com;tag=CDASE-123134ad
Call-ID:BW15331149901048856990 5@10.16.134.17
CSeq:29 UPDATE
Contact:<sip:as1.broadsoft.com:5060>;src
Supported:100rel
Allow:ACK,BYE,CANCEL,INFO,INVITE,OPTIONS,PRACK,REFER,NOTIFY,UPDATE
Accept:application/media_control+xml,application/sdp,multipart/mixed
Max-Forwards:10
Content-Type: application/rs-metadata+xml
Content-Disposition: recording-session
```

```
Content-Length: ...

<?xml version="1.0" encoding="UTF-8"?>
<recording-metadata xmlns="urn:ietf:params:xml:ns:recording"
id="urn:uuid:34512345-6743-6248-9043897645ab">
  <dataMode>partial</dataMode>
  <extensionData id="urn:uuid:ef45678456-4451-4568-7785-400554586487"
               parent="urn:uuid:78554655-7844-5564-4568-ef4566246875">
    <broadWorksRecordingMetadata
xmlns="http://schema.broadsoft.com/broadworksCallRecording"
schemaRev="1.0"
xsi:schemaLocation="http://schema.broadsoft.com/broadworksCallRecording"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <extTrackingID>5d81bb9f-c34d231@10.16.134.17</extTrackingID>
      <redirectCallInfo>
        <redirectedFromPartyNumber>9725551212@broadsoft.com
        </redirectedFromPartyNumber>
        <redirectedToPartyNumber>9725551414@broadsoft.com
        </redirectedToPartyNumber>
        <\redirectCallInfo>
    </broadWorksRecordingMetadata>
  </extensionData>
</recording-metadata>
```

9.7.5 Example NOTIFY with CallRecordingModeEvent

The following example shows a NOTIFY with the new *CallRecordingModeEvent.* The provisioned call recording mode values can be “always”, “always-pause-resume”, “on-demand”, “on-demand-user-start”, or “never”.

The NOTIFY’s message body is an XML body with a MIME content-type of *application/x-as-feature-event+xml*.

```
NOTIFY sip:10.17.0.133:5060 SIP/2.0
Via:SIP/2.0/UDP 10.17.0.120;branch=z9hG4bKBroadWorks.213gmu-
10.17.0.133V5060-0-978860575-1687133437-1365704718875
From:<sip:5154752162@as.bw5.rtx.broadsoft.com>;tag=1687133437-
1365704718875
To:<sip:5154752162@as.bw5.rtx.broadsoft.com>;tag=769449388-1365704718873-
Call-ID:BW1325188731104131014012462@AMSSignalingGateway
CSeq:978860575 NOTIFY
Contact:<sip:as.bw5.rtx.broadsoft.com:5060>
Event:as-feature-event
Subscription-State:active;expires=3599
Max-Forwards:5
Content-Type:application/x-as-feature-event+xml
Content-Length:271

<?xml version="1.0" encoding="UTF-8"?><xBw:CallRecordingModeEvent
xmlns:xBw="http://schema.broadsoft.com/as-feature-
event"><xBw:callRecordingMode>always</xBw:callRecordingMode></xBw:CallRec
ordingModeEvent>
```

9.7.6 Example INVITE with "record-aware" in Supported Header

Recording-awareness is indicated by the presence of the "record-aware" option tag in the *Supported* header in the initial INVITE request or response.

```
INVITE sip:501@10.16.150.2 SIP/2.0
Via: SIP/2.0/UDP 10.16.150.100;branch=z9hG4bK9072197a7a12d36a
From: "north00" <sip:9726996500@10.16.150.2>;tag=0fa57cd3cd49f777
To: <sip:501@10.16.150.2>
Contact: <sip:9726996500@10.16.150.100>
Supported: replaces,record-aware
Call-ID: 99e8ab192ef14478@10.16.150.100
CSeq: 59967 INVITE
User-Agent: Grandstream BT100 1.0.5.23
Max-Forwards: 70
Allow: INVITE,ACK,CANCEL,BYE,NOTIFY,REFER,OPTIONS,INFO,SUBSCRIBE
Content-Type: application/sdp
Content-Length: 416

v=0
o=9726996500 8000 8000 IN IP4 10.16.150.100
s=SIP Call
c=IN IP4 10.16.150.100
t=0 0
m=audio 5004 RTP/AVP 8 0 4 18 2 15 99 9 101
a=sendrecv
a=rtpmap:8 PCMA/8000
a=rtpmap:0 PCMU/8000
a=rtpmap:4 G723/8000
a=rtpmap:18 G729/8000
a=rtpmap:2 G726-32/8000
a=rtpmap:15 G728/8000
a=rtpmap:99 iLBC/8000
a=fmtp:99 mode=20
a=rtpmap:9 G722/16000
a=ptime:20
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-11
```

9.7.7 Example SDP with *a=record* Attribute

The status of the recording is indicated by a new *a=record* SDP attribute.

```
ACK sip:9726996500@10.16.150.100 SIP/2.0
Via:SIP/2.0/UDP 10.16.150.2;branch=z9hG4bKBroadWorks.16sncpm-
10.16.150.100V5060-0-1064909701A165295465-1365781936461
From:<sip:501@10.16.150.2>;tag=165295465-1365781936461
To:"north00"<sip:9726996500@10.16.150.2>;tag=0fa57cd3cd49f777
Call-ID:99e8ab192ef14478@10.16.150.100
CSeq:1064909701 ACK
Contact:<sip:10.16.150.2:5060>
Max-Forwards:10
Content-Type:application/sdp
Content-Length:137

v=0
o=BroadWorks 8 1 IN IP4 10.16.120.22
s=-
c=IN IP4 10.16.120.22
```

```
t=0 0
m=audio 18748 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=ptime:20
a=record:on
```

9.7.8 Example SDP with *a=recordpref* Attribute

To change the state of the recording, the recording preference is indicated by the *a=recordpref* attribute in the SDP.

```
INVITE sip:10.16.150.2:5060 SIP/2.0
Via: SIP/2.0/UDP 10.16.150.100;branch=z9hG4bK276c6dc4fa4c1854
From: "north00" <sip:9726996500@10.16.150.2>;tag=6a02e73078a5ffc3
To: <sip:501@10.16.150.2>;tag=1084150047-1365624758806
Contact: <sip:9726996500@10.16.150.100>
Supported: replaces
Call-ID: a34d4675970efcc6@10.16.150.100
CSeq: 62413 INVITE
User-Agent: Grandstream BT100 1.0.5.23
Max-Forwards: 70
Allow: INVITE,ACK,CANCEL,BYE,NOTIFY,REFER,OPTIONS,INFO,SUBSCRIBE
Content-Type: application/sdp
Content-Length: 360

v=0
o=9726996500 8000 8002 IN IP4 10.16.150.100
s=SIP Call
c=IN IP4 10.16.150.100
t=0 0
m=audio 5004 RTP/AVP 8 0 4 18 2 15 99 9
a=sendrecv
a=rtpmap:8 PCMA/8000
a=rtpmap:0 PCMU/8000
a=rtpmap:4 G723/8000
a=rtpmap:18 G729/8000
a=rtpmap:2 G726-32/8000
a=rtpmap:15 G728/8000
a=rtpmap:99 iLBC/8000
a=fmtp:99 mode=20
a=rtpmap:9 G722/16000
a=ptime:20
a=recordpref:on
```

10 Call Flow Details

This section shows some of the basic call flows. In the call flows, the 100 trying messages are skipped for the sake of brevity. For this same reason, only headers that are important to the call flows are shown. This means some mandatory SIP headers are missing. The call flows highlight the headers that are important for the interface to the Call Recording platform. For more information on the XML schema for the *BroadWorks-Recording-metadata*, see section *9.6.2 broadWorksRecordingMetadata*.

10.1 Originator Recording Call

This call flow shows the messages when the originator has Call Recording enabled. The originator is using a single stream to the Call Recording Server. Details of the messages to/from the Call Recording platform appear after the call flow diagram.

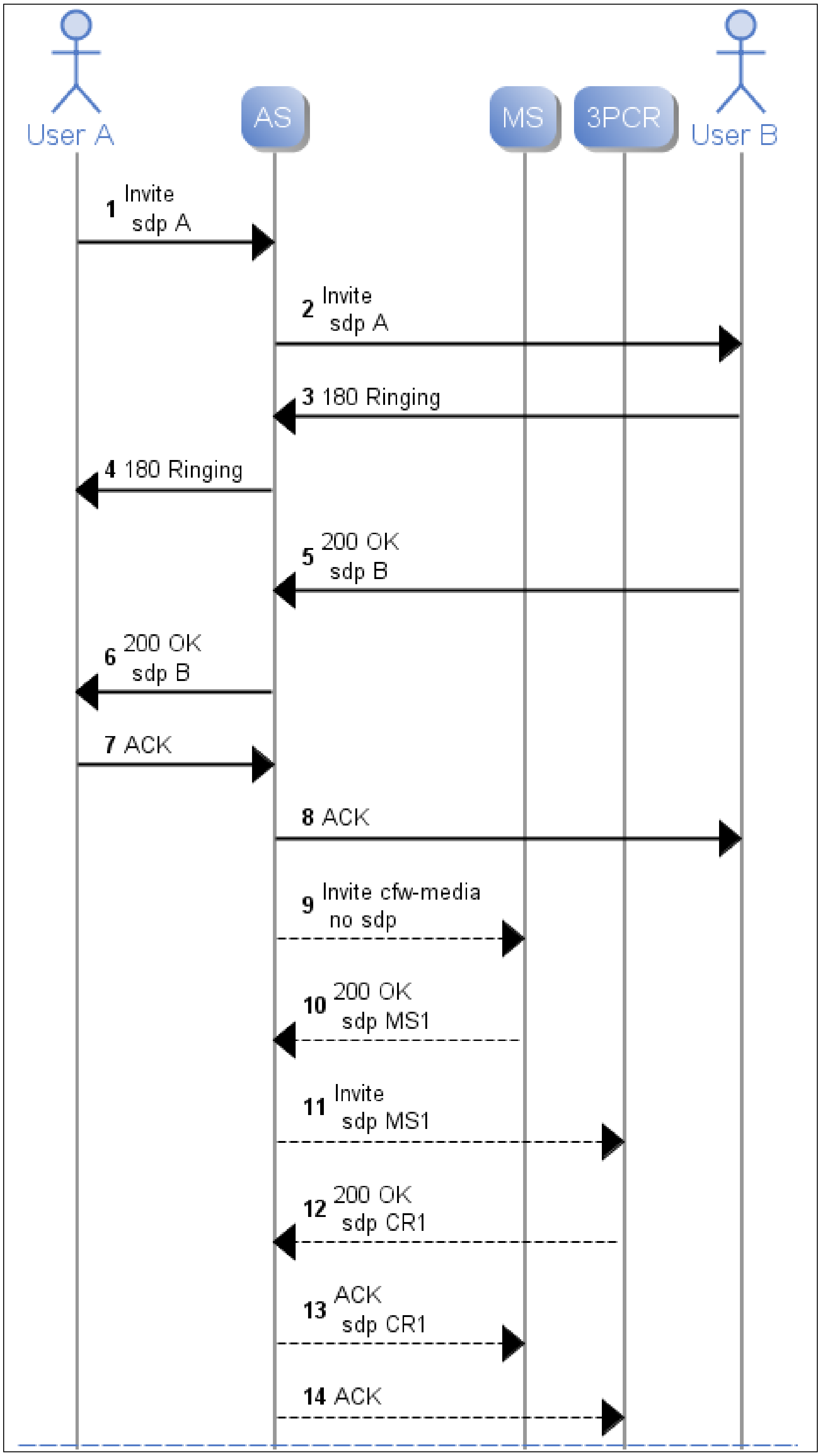

Figure 15 Originator Records Call (a)

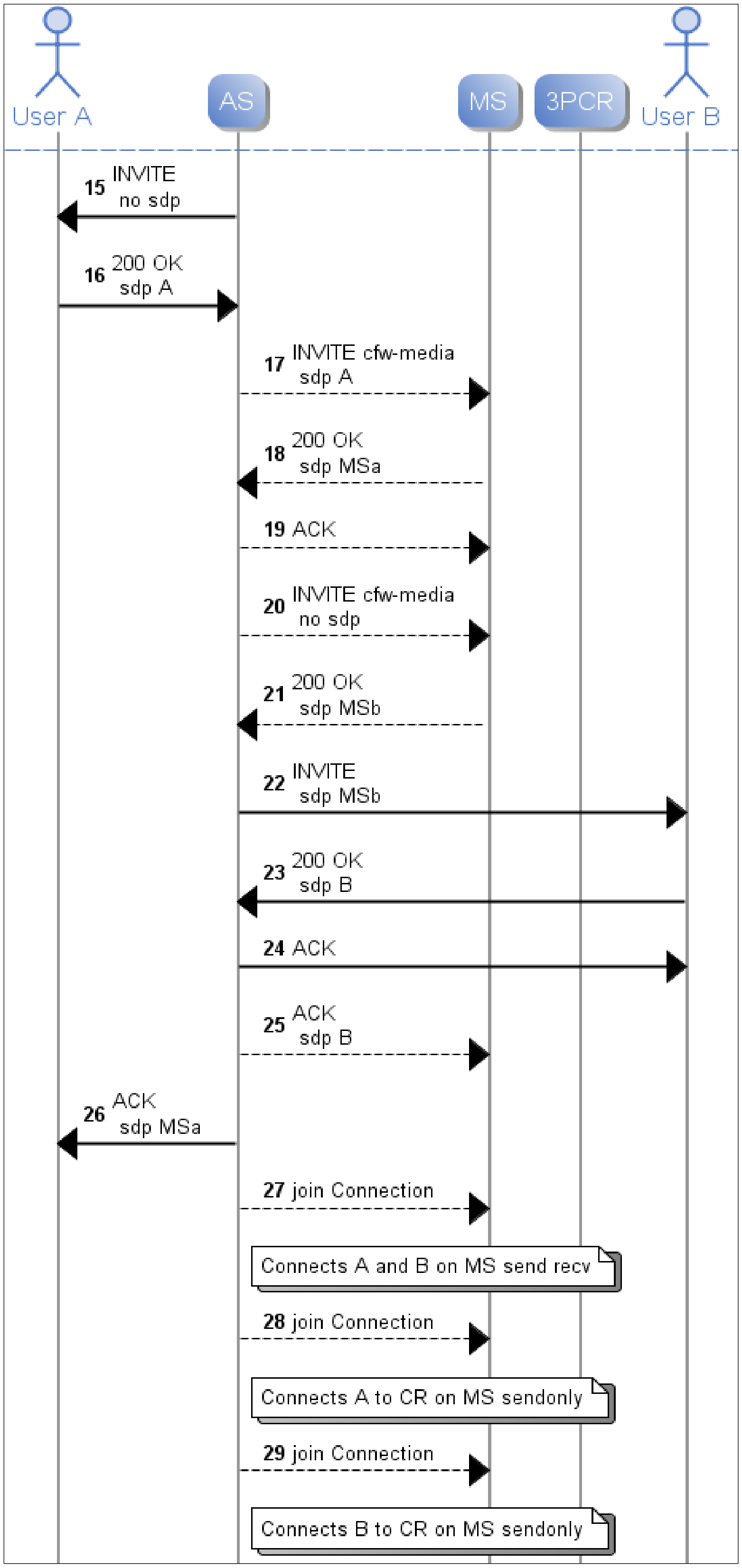

Figure 16 Originator Records Call (b)

10.1.1 INVITE to Call Recording Platform

This is an example INVITE sent to the Call Recording platform. Note the inclusion of the *src* parameter in the *Contact* header and the empty *Recv-Info* header. Additionally, the message body is a multipart body consisting of the SDP description and the information about the call recording is included in the Cisco BroadWorks recording metadata.

```
INVITE sip:3PCR@recorder.broadsoft.com:5060;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26ff4f26931B5DED
From: <sip:as1.broadsoft.com>;tag=B6CB9EB1-8AE7B5DC
To: sip:3PCR@recorder.broadsoft.com
Call-ID:BW153311499010411112159050@10.16.134.17
CSeq:157450406 INVITE
Contact:<sip:as1.broadsoft.com:5060>;src
Supported:100rel
Recv-Info:
Allow:ACK,BYE,CANCEL,INFO,INVITE,OPTIONS,PRACK,REFER,NOTIFY,UPDATE
Accept:application/media_control+xml,application/sdp,multipart/mixed
Max-Forwards:10
Content-Type: multipart/mixed;boundary=dfe2341adf13412fqdfadq
Content-Length: ...

--dfe2341adf13412fqdfadq
Content-Type: application/sdp

v=0
o=BroadWorks 17 1 IN IP4 10.16.134.101
s=-
c=IN IP4 10.16.134.101
t=0 0
m=audio 2262 RTP/AVP 8
a=rtpmap:8 PCMA/8000
a=sendonly
a=label:1

--dfe2341adf13412fqdfadq
Content-Type: application/rs-metadata+xml
Content-Disposition: recording-session

<?xml version="1.0" encoding="UTF-8"?>
<recording-metadata xmlns="urn:ietf:params:xml:ns:recording"
id="urn:uuid:1431348789-13413-13414-134">
  <dataMode>complete</dataMode>
  <recording id="urn:uuid:12345678-1234-1234-1234-123456789012">
    <requestor>src</requestor>
  </recording>
  <group id="urn:uuid:98736524-ft12-891p-3pui-156ghesRTDFgh"
         recording="urn:uuid:12345678-1234-1234-1234-123456789012"/>
  <session id="urn:uuid:ui129933-4524-2542-6603-13414314duw0"
           group="urn:uuid:98736524-ft12-891p-3pui-156ghesRTDFgh">
  </session>
  <stream id="urn:uuid:94dwif31-kl2w-341d-12id-duUHawq12348"
          session="urn:uuid:ui129933-4524-2542-6603-13414314duw0">
    <label>1</label>
    <mode>mixed</mode>
  </stream>
  <extensionData id="urn:uuid:4253409701-0097134113-1398791834"
                 parent="urn:uuid:ui129933-4524-2542-6603-13414314duw0">
```

```
    <broadWorksRecordingMetadata
xmlns="http://schema.broadsoft.com/broadworksCallRecording"
schemaRev="1.0"
xsi:schemaLocation="http://schema.broadsoft.com/broadworksCallRecording"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <extTrackingID>2:1</extTrackingID>
      <serviceProviderID>TxASDev</serviceProviderID>
      <groupID>North_as90</groupID>
      <userID>north00@rtx.broadsoft.com</UserID>
      <callID> BW1533114111215905@10.16.134.17</CallID>
      <callType>
        <origCall>

<callingPartyNumber>9725551212@broadsoft.com</callingPartyNumber>
          <calledPartyNumber>2146412365</calledPartyNumber>
          <dialedDigits>2146412365</dialedDigits>
        </origCall>
      </callType>
      <recordingType>on</recordingType>
    </broadWorksRecordingMetadata>
  </extensionData>
</recording-metadata>
--dfe2341adf13412fqdfadq—
```

10.1.2 200 OK from Call Recording Platform

This is the *200 OK* returned by the Call Recording platform. Note that it includes the *Recv-Info* header and that the *Contact* header contains the *srs* parameter.

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26ff4f26931B5DED
From: <sip:as1.broadsoft.com>;tag=B6CB9EB1-8AE7B5DC
To: sip:3PCR@recorder.broadsoft.com;tag=31234134987913
Call-ID:BW1533114990104111215905@10.16.134.17
CSeq:157450406 INVITE
Contact:<sip:recorder.broadsoft.com:5060>;srs
Recv-Info: x-broadworks-callrecording
Allow:ACK,BYE,CANCEL,INFO,INVITE,OPTIONS,PRACK,REFER,NOTIFY,UPDATE
Content-Type: application/sdp
Content-Length: ...

v=0
o=BroadWorks 17 1 IN IP4 10.16.134.110
s=-
c=IN IP4 10.16.134.110
t=0 0
m=audio 26584 RTP/AVP 8
a=rtpmap:8 PCMA/8000
a=recvonly
a=label:1
```

10.1.3 ACK to Call Recording Platform

This is an example of the ACK sent to the Call Recording platform. Note the inclusion of the *src* parameter in the *Contact* header.

```
ACK sip:recorder.broadsoft.com SIP/2.0
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26ff4f2693131431
From: <sip:as1.broadsoft.com>;tag=B6CB9EB1-8AE7B5DC
```

```
To: sip:3PCR@recorder.broadsoft.com;tag=31234134987913
Call-ID:BW15331149901041111215905@10.16.134.17
CSeq:157450406 ACK
Contact:<sip:as1.broadsoft.com:5060>;src
Content-Length: 0
```

10.2 Start, Stop, Pause, and Resume

10.2.1 On Demand Keep, Then Pause Recording

The following call flow picks up from the originator recording the call case. This flow starts after message 25 and shows the user initiating the Xtended Services Interface messaging to keep the recording and then the flow for pausing and resuming the recording.

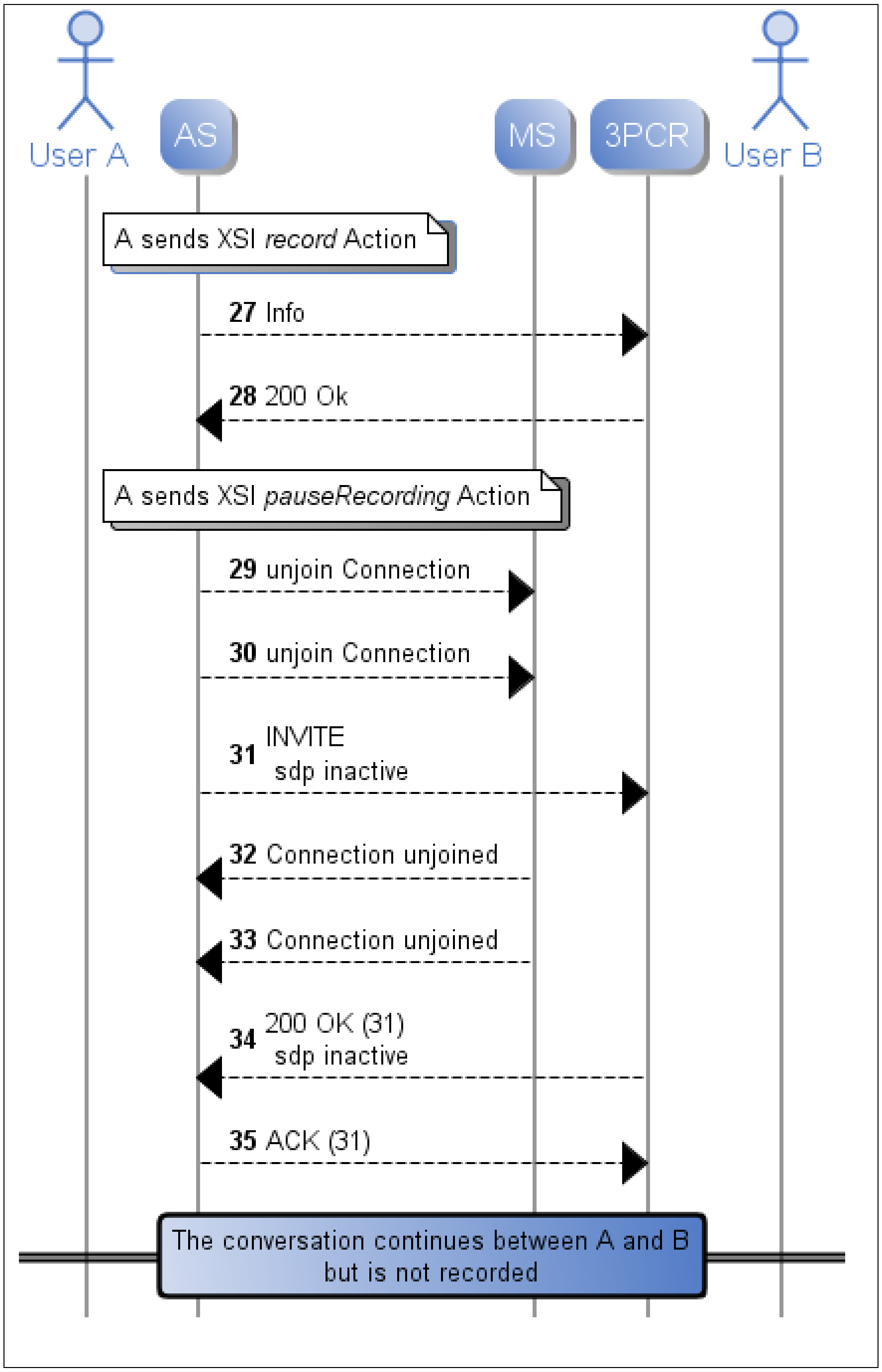

Figure 17 Pause of On-Demand Call

10.2.1.1 INFO Message

Following is an example of the INFO message sent by the Application Server to the 3PCR platform. There is no message body in the INFO. The presence of the *Info-Package* and the fact that the INFO message is part of the ongoing dialog for the call recording provide sufficient information to identify the call recording to be kept. The Application Server does not send a message body in the INFO.

```
INFO sip:recorder.broadsoft.com SIP/2.0
```

```
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26ff4f44887554
From: <sip:as1.broadsoft.com>;tag= B6CB9EB1-8AE7B75C
To: sip:3PCR@recorder.broadsoft.com;tag=45DEF34123412DF
Call-ID: BW153311499010488569905@10.16.134.17
CSeq:26 INFO
Contact:<sip:as1.broadsoft.com:5060>;src
Max-Forwards: 70
Info-Package: x-broadworks-callrecording
Content-Length: 0
```

10.2.1.2 INVITE Message

Message 31 is sent in response to the Xtended Services Interface *pauserecording* event. All of the media streams are set to “a=inactive”. The INVITE is sent to both the 3PCR platform and the Media Server to pause the recording of the call. This only affects the recording of the call and not the call between users.

The following INVITE is an example of what can be sent to the 3PCR platform. There is no change to the metadata for pausing the recording; therefore, there is no metadata in the INVITE.

```
INVITE sip:10.16.134.17:5070;transport=UDP SIP/2.0
Via:SIP/2.0/UDP 10.16.134.17;branch=z9hG4bKBroadWorks.1lp0lgs-
10.16.134.17V5070-0-779834173-1850244762-1352326882930-
From:"Scott
North"<sip:9726990600@rtx.broadsoft.com;user=phone>;tag=1850244762-
1352326882930-
To:<sip:10.16.134.17:5070>;tag=4
Call-ID:BW162122930071112141407536 9@10.16.134.17
CSeq:779834173 INVITE
Route:<sip:10.16.134.17:5070;lr>
Contact:<sip:10.16.134.17:5060>;src
Max-Forwards:10
Content-Type:application/sdp
Content-Length:627

v=0
o=BroadWorks 141 1 IN IP4 10.16.120.22
s=-
c=IN IP4 0.0.0.0
t=0 0
m=audio 12348 RTP/AVP 0 8 18 96 9 2 101
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:96 AMR/8000
a=rtpmap:9 G722/8000
a=rtpmap:2 G726-32/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=ptime:20
a=label:1
a=inactive
```

10.2.2 On Demand Resume Recording

The following call flow shows the user resuming the call recording. It picks up from message 33 in the pause call flow. User A initiates the “resume” of the recording using the Xtended Services Interface *resumeRecording* event.

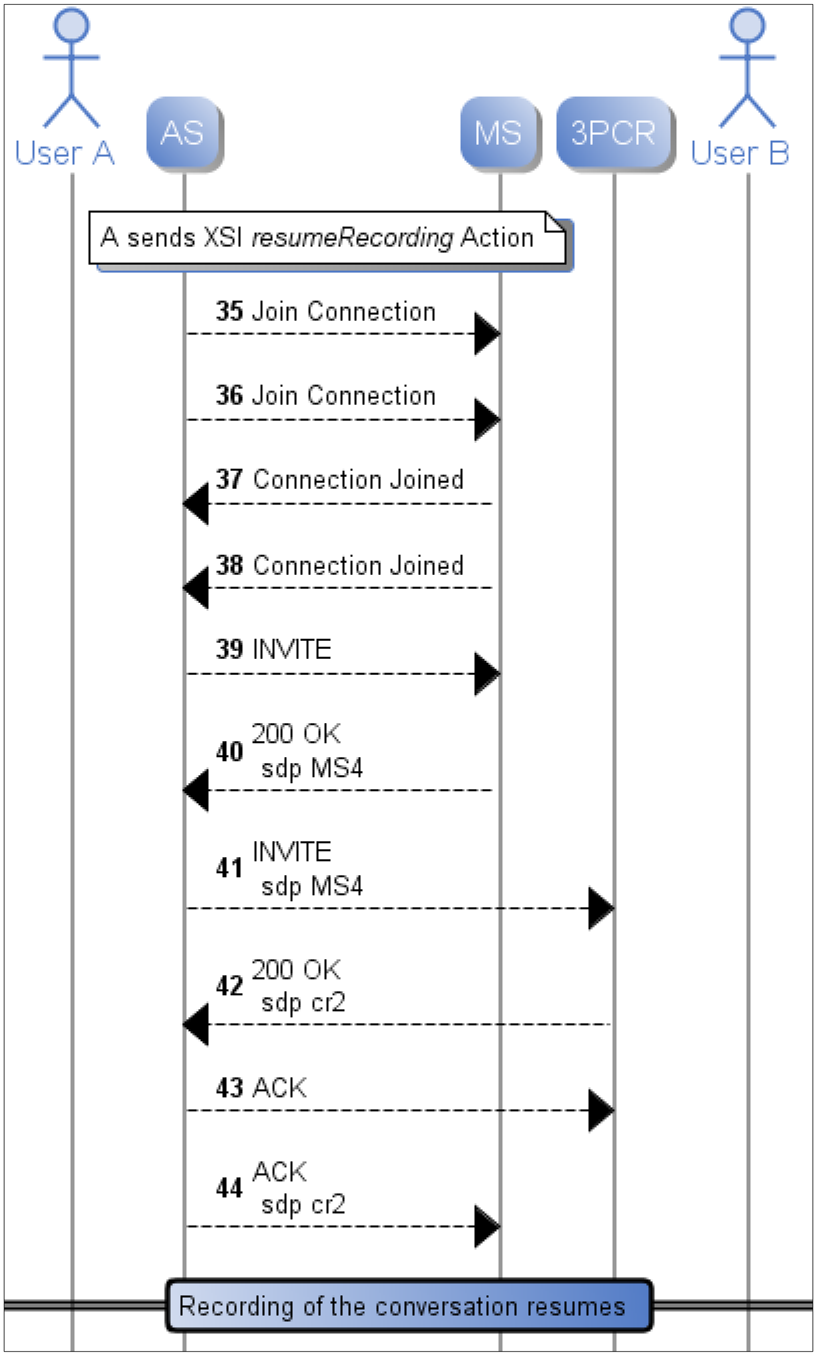

Figure 18 Resume On-Demand Call

10.2.2.1 INVITE to Media Server

Right after the "resume" event, the first INVITE sent to the Media Server does not contain the SDP. This causes the Media Server to return its full offer SDP capabilities in the *200 OK* response to the INVITE. The SDP from the Media Server is used to re-INVITE the 3PCR platform.

10.2.2.2 INVITE to 3PCR Platform

On the “resume”, the INVITE to the 3PCR platform contains the full SDP of the Media Server to restart the recording. There is no change in the metadata; therefore, it is not included in the message.

```
INVITE sip:3PCR@recorder.broadsoft.com:5060;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26f33f26931B5DED
From: <sip:as1.broadsoft.com>;tag=B6CB9EB1-8AE7B75C
To: sip:3PCR@recorder.broadsoft.com
Call-ID:BW153311499010488569905@10.16.134.17
CSeq:25 INVITE
Contact:<sip:as1.broadsoft.com:5060>;src
Supported:100rel
Recv-Info:
Allow:ACK,BYE,CANCEL,INFO,INVITE,OPTIONS,PRACK,REFER,NOTIFY,UPDATE
Accept:application/media_control+xml,application/sdp,multipart/mixed
Max-Forwards:10
Content-Type: application/sdp
Content-Length: ...

v=0
o=BroadWorks 1141341234 131413 IN IP4 10.16.134.101
s=-
c=IN IP4 10.16.134.101
t=0 0
m=audio 15096 RTP/AVP 0 18 9
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=label:1
```

10.2.3 On Demand with User Initiated Start

The call flows in this section start with calls that are already in session. The call flows cover:

- Starting a recording mid-call
- Multiple recordings on the same call
- Conference call:
  - Starting after the conference is established.
  - Starting the conference when only the call between User A and User B is being recorded.
  - Adding a call that is being recorded to a conference that is not being recorded.
  - Adding a call that is being recorded to a conference that is being recorded.

10.2.4 Mid-Call Start

In this call flow, there is an initial call between User A and User B. User A decides to record the call and sends in the Xtended Services Interface *record* event. This event causes the Application Server to add in the 3PCR platform and connect all of the parties to the call recording.

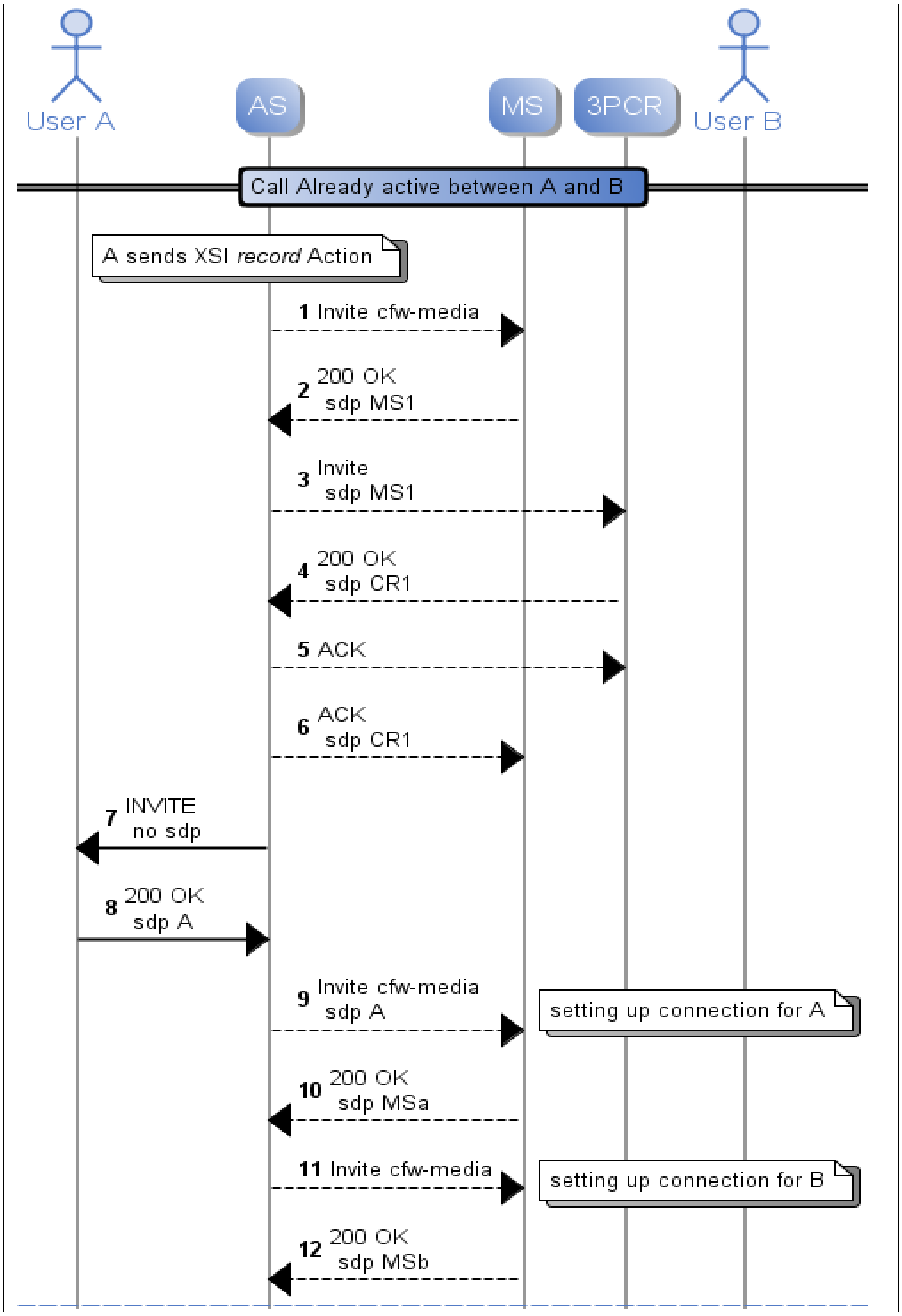

Figure 19 Mid-Call Start (a)

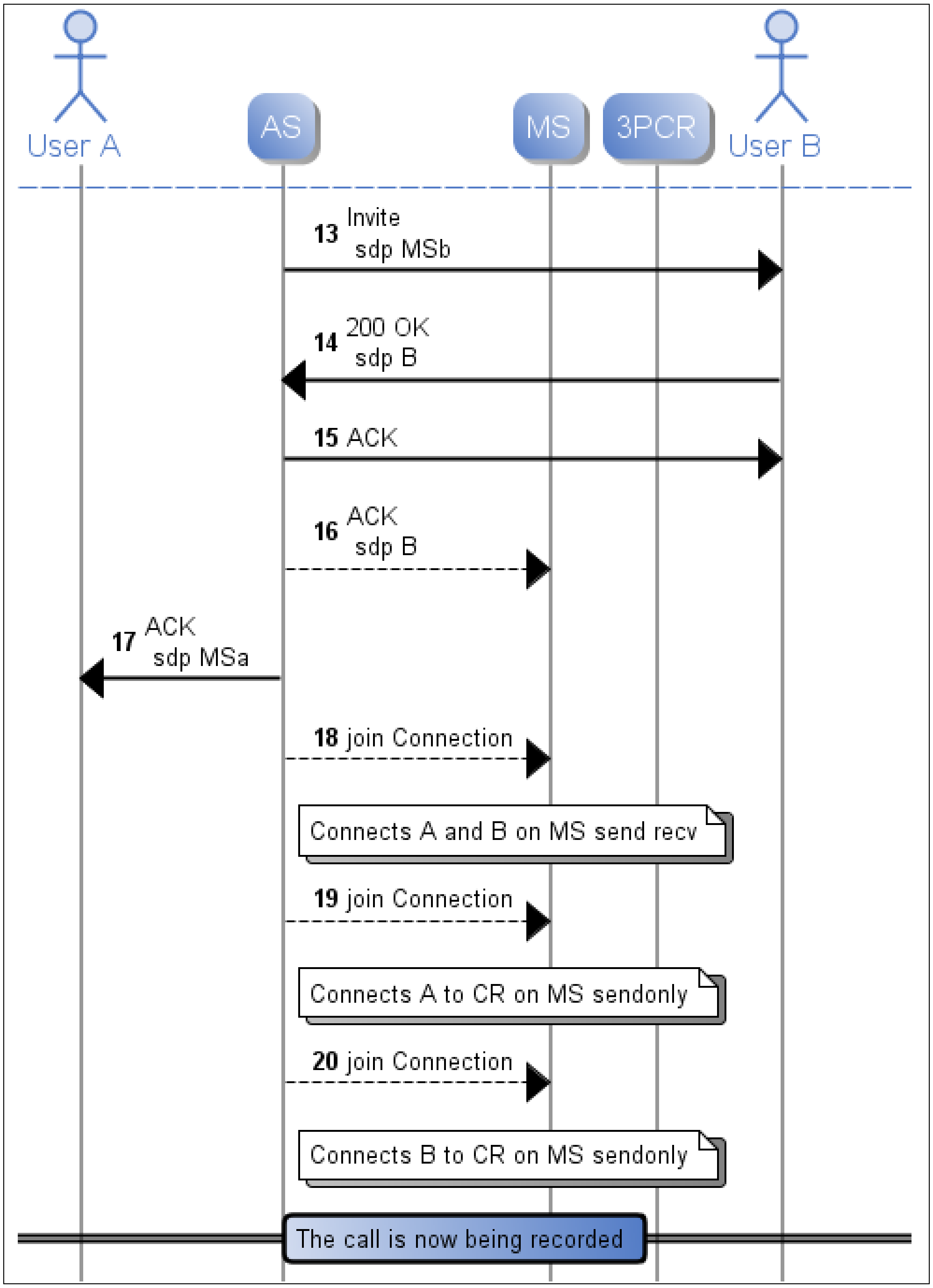

Figure 20 Mid-Call Start (b)

10.2.4.1 INVITE to Third-Party Call Recording Platform

Following is an example INVITE sent to the 3PCR platform. Note the inclusion of the *src* parameter in the *Contact* header and the empty *Recv-Info* header. Additionally, the message body is a multipart body consisting of the SDP description. The information regarding the call recording is included in the Cisco BroadWorks recording metadata.

```
INVITE sip:3PCR@recorder.broadsoft.com:5060;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26ff4f26931B5DED
```

```
From: <sip:as1.broadsoft.com>;tag=B6CB9EB1-8AE7B5DC
To: sip:3PCR@recorder.broadsoft.com
Call-ID:BW1533114990104111121590 5@10.16.134.17
CSeq:157450406 INVITE
Contact:<sip:as1.broadsoft.com:5060>;src
Supported:100rel
Recv-Info:
Allow:ACK,BYE,CANCEL,INFO,INVITE,OPTIONS,PRACK,REFER,NOTIFY,UPDATE
Accept:application/media_control+xml,application/sdp,multipart/mixed
Max-Forwards:10
Content-Type: multipart/mixed;boundary=abc21231adfa134adf141378
Content-Length: ...

--abc21231adfa134adf141378
Content-Type: application/sdp

v=0
o=BroadWorks 18 1 IN IP4 10.16.134.101
s=-
c=IN IP4 10.16.134.101
t=0 0
m=audio 15096 RTP/AVP 0 18 9
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=label:1


--abc21231adfa134adf141378
Content-Type: application/rs-metadata+xml
Content-Disposition: recording-session

<?xml version="1.0" encoding="UTF-8"?>
<recording-metadata xmlns="urn:ietf:params:xml:ns:recording"
id="urn:uuid:1431348789-13413-13414-134">
  <dataMode>complete</dataMode>
  <recording id="urn:uuid:12345678-1234-1234-1234-123456789012">
    <requestor>src</requestor>
  </recording>
  <group id="urn:uuid:98736524-ft12-89lp-3pui-156ghesRTDFgh"
         recording="urn:uuid:12345678-1234-1234-1234-123456789012"/>
  <session id="urn:uuid:ui129933-4524-2542-6603-13414314duw0"
           group="urn:uuid:98736524-ft12-89lp-3pui-156ghesRTDFgh">
  </session>
  <stream id="urn:uuid:94dwif31-kl2w-341d-12id-duUHawq12348"
          session="urn:uuid:ui129933-4524-2542-6603-13414314duw0">
    <label>1</label>
    <mode>mixed</mode>
  </stream>
  <extensionData id="urn:uuid:4253409701-0097134113-1398791834"
                 parent="urn:uuid:ui129933-4524-2542-6603-13414314duw0">
    <broadWorksRecordingMetadata
xmlns="http://schema.broadsoft.com/broadworksCallRecording"
schemaRev="1.0"
xsi:schemaLocation="http://schema.broadsoft.com/broadworksCallRecording"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <extTrackingID>2:1</extTrackingID>
      <serviceProviderID>TxASDev</serviceProviderID>
      <groupID>North_as90</groupID>
      <userID>north00@rtx.broadsoft.com</UserID>
      <callID> BW1533114111215905@10.16.134.17</CallID>
```

```
      <callType>
        <origCall>

<callingPartyNumber>9725551212@broadsoft.com</callingPartyNumber>
          <calledPartyNumber>2146412365</calledPartyNumber>
          <dialedDigits>2146412365</dialedDigits>
        </origCall>
      </callType>
      <recordingType>on</recordingType>
    </broadWorksRecordingMetadata>
  </extensionData>
</recording-metadata>
--abc21231adfa134adf141378—
```

10.2.5 Mid-Call Stop

The following call flow shows the mid-call stop of a call that is currently being recorded. This call flow continues just after the call flow in *Figure 20*.

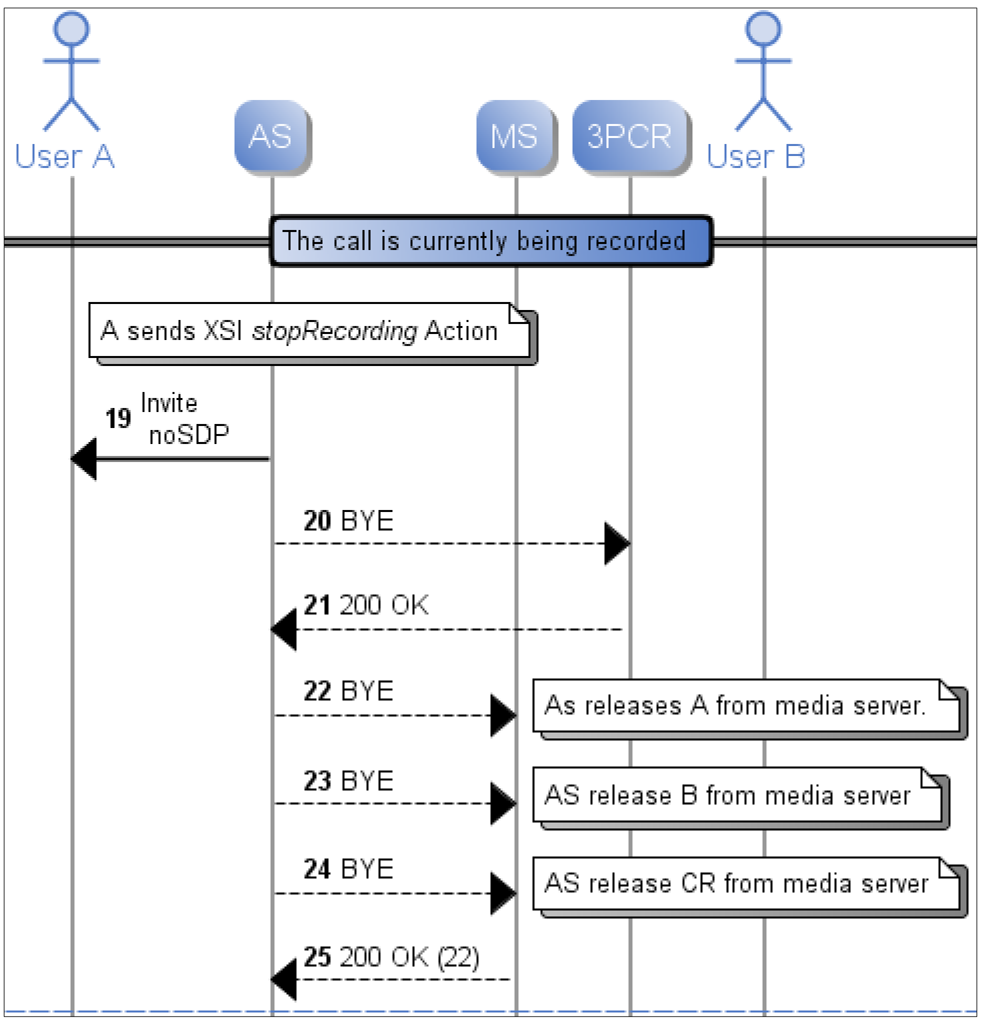

Figure 21 Mid-Call Stop (a)

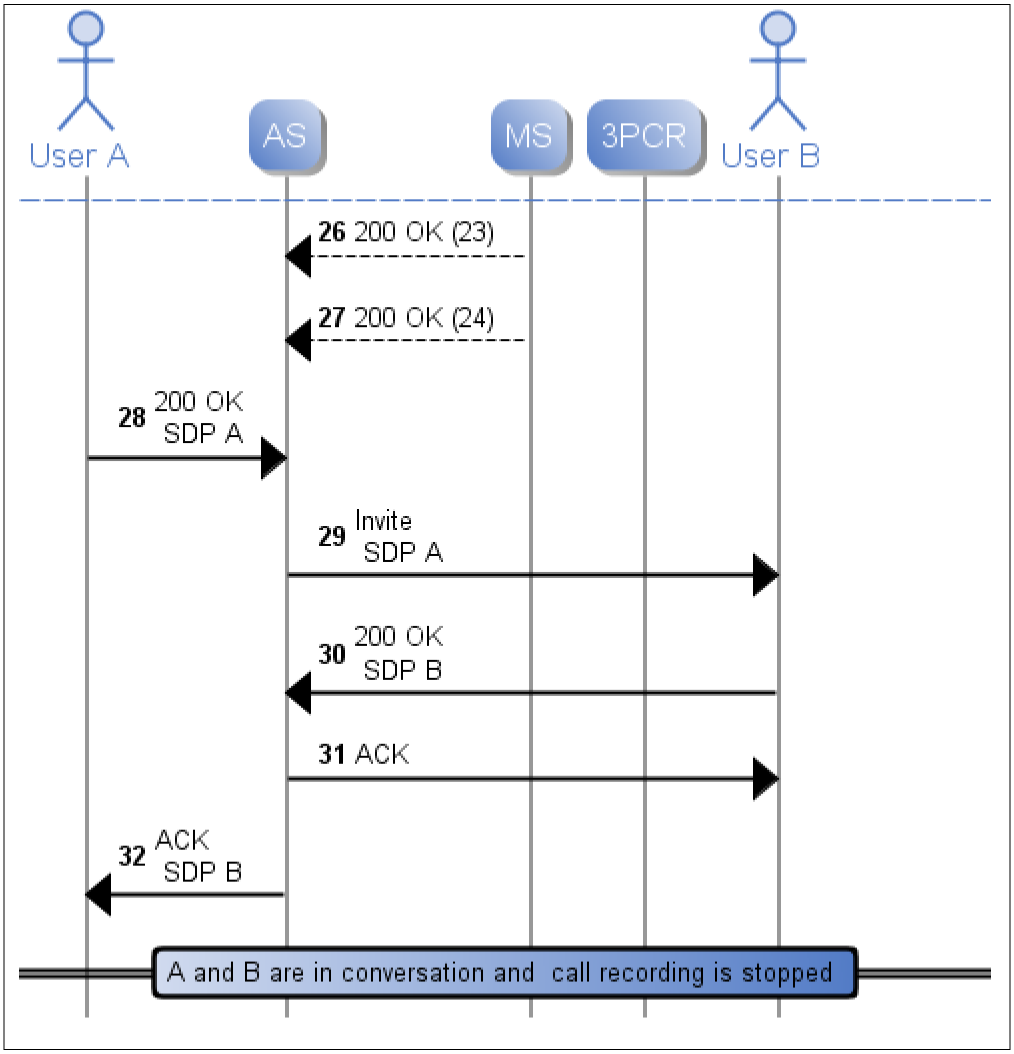

Figure 22 Mid-Call Stop (b)

10.2.6 Multiple Recording on Same Call

The following call flow shows a situation in which a user who has previously recorded a call stops the recording and decides to start recording again. This call flow picks up from the recording shown in *Figure 22*.

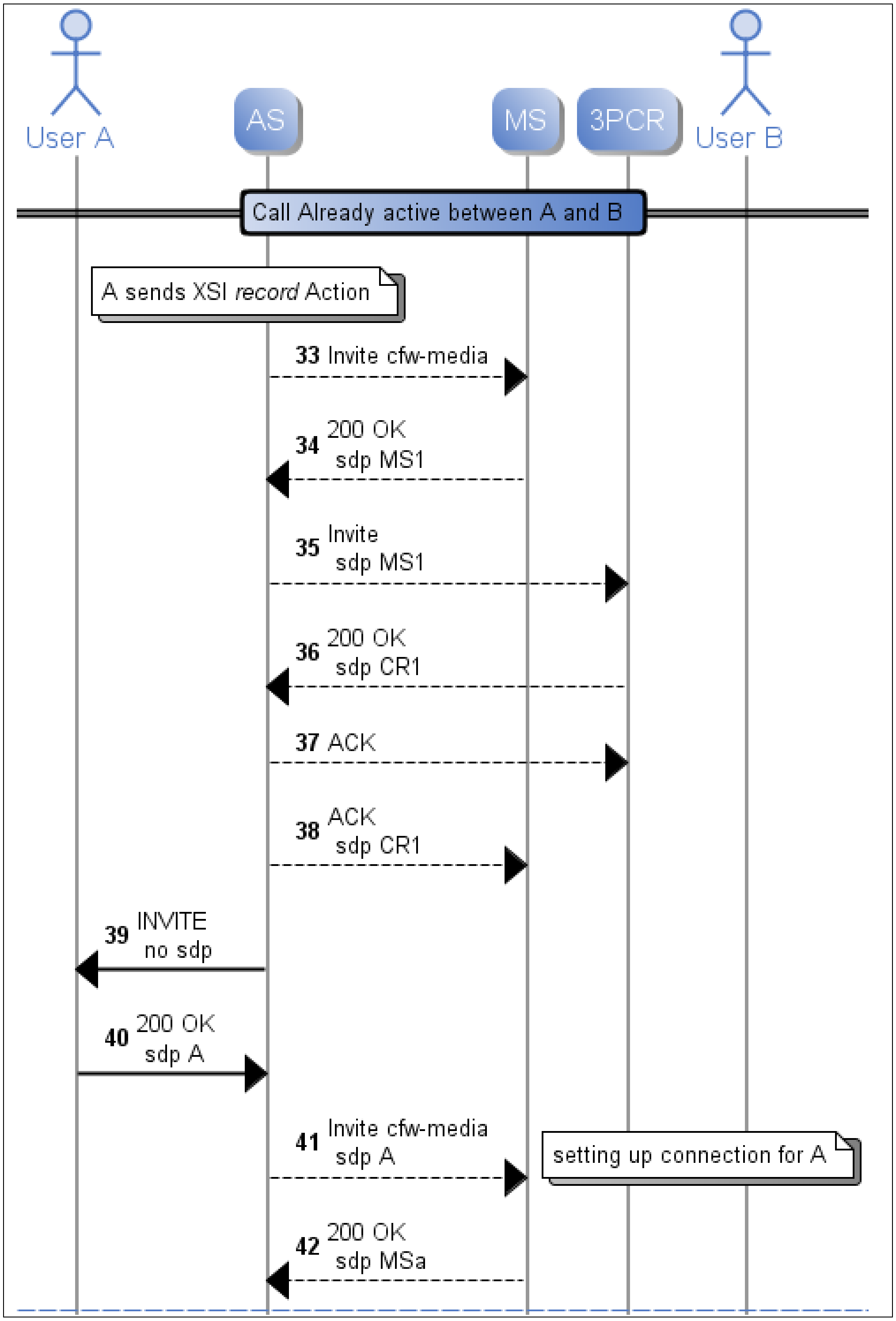

Figure 23 Mid-Call Restart (a)

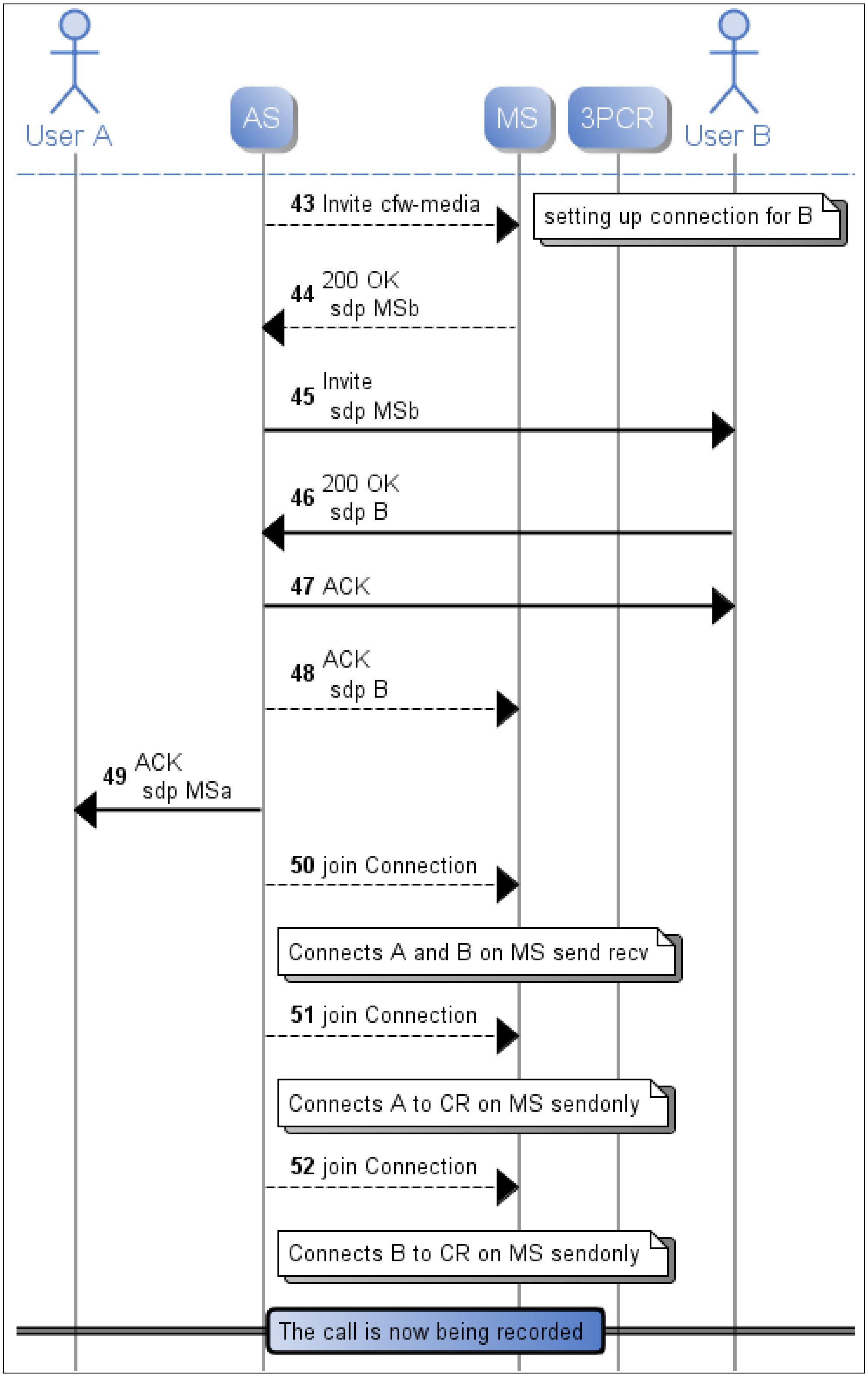

Figure 24 Mid-Call Restart (b)

10.2.6.1 INVITE to 3PCR Platform

Following is an example INVITE sent to the 3PCR platform. Note the inclusion of the *src* parameter in the *Contact* header and the empty *Recv-Info* header. Additionally, the message body is a multipart body consisting of the SDP description. The information regarding the call recording is included in the Cisco BroadWorks recording metadata.

```
INVITE sip:3PCR@recorder.broadsoft.com:5060;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26ff4f26931B5DED
From: <sip:as1.broadsoft.com>;tag=B6CB9EB1-8AE7B5DC
To: sip:3PCR@recorder.broadsoft.com
Call-ID:BW153311499010411112159O5@10.16.134.17
CSeq:157450410 INVITE
Contact:<sip:as1.broadsoft.com:5060>;src
Supported:100rel
Recv-Info:
Allow:ACK,BYE,CANCEL,INFO,INVITE,OPTIONS,PRACK,REFER,NOTIFY,UPDATE
Accept:application/media_control+xml,application/sdp,multipart/mixed
Max-Forwards:10
Content-Type: multipart/mixed;boundary=kafdads314315f870qreq
Content-Length: ...

--kafdads314315f870qreq
Content-Type: application/sdp

v=0
o=BroadWorks 19 1 IN IP4 10.16.134.101
s=-
c=IN IP4 10.16.134.101
t=0 0
m=audio 2262 RTP/AVP 8
a=rtpmap:8 PCMA/8000
a=sendonly
a=label:1

--kafdads314315f870qreq
Content-Type: application/rs-metadata+xml
Content-Disposition: recording-session

<?xml version="1.0" encoding="UTF-8"?>
<recording-metadata xmlns="urn:ietf:params:xml:ns:recording"
id="urn:uuid:1431348789-17151-13414-134">
  <dataMode>complete</dataMode>
  <recording id="urn:uuid:12345678-7878-1234-1234-123456789012">
    <requestor>src</requestor>
  </recording>
  <group id="urn:uuid:98736524-9879-89lp-3pui-156ghesRTDFgh"
         recording="urn:uuid:12345678-787-1234-1234-123456789012"/>
  <session id="urn:uuid:ui129933-1594-2542-6603-13414314duw0"
           group="urn:uuid:98736524-9879-89lp-3pui-156ghesRTDFgh">
  </session>
  <stream id="urn:uuid:94dwif31-kl2w-341d-12id-duUHawq12348"
          session="urn:uuid:ui129933-1594-2542-6603-13414314duw0">
    <label>1</label>
    <mode>mixed</mode>
  </stream>
  <extensionData id="urn:uuid:4253409701-0097134113-1398791834"
                 parent="urn:uuid:ui129933-1594-2542-6603-13414314duw0">
```

```
    <broadWorksRecordingMetadata
xmlns="http://schema.broadsoft.com/broadworksCallRecording"
schemaRev="1.0"
xsi:schemaLocation="http://schema.broadsoft.com/broadworksCallRecording"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <extTrackingID>2:1</extTrackingID>
      <serviceProviderID>TxASDev</serviceProviderID>
      <groupID>North_as90</groupID>
      <userID>north00@rtx.broadsoft.com</UserID>
      <callID> BW1533114111215905@10.16.134.17</CallID>
      <callType>
        <origCall>

<callingPartyNumber>9725551212@broadsoft.com</callingPartyNumber>
          <calledPartyNumber>2146412365</calledPartyNumber>
          <dialedDigits>2146412365</dialedDigits>
        </origCall>
      </callType>
      <recordingType>on</recordingType>
    </broadWorksRecordingMetadata>
  </extensionData>
</recording-metadata>
--kafdads314315f870qreq—
```


10.2.7 Conference Call; Start After Conference Established

In the following figure, the conference call has already been established between Users A, B, and C. User C is the conference controller and User C starts recording the conference call. Note that in this case, the call recording is set up between the controller and the conference bridge. This example uses the dual recording mode so there are two streams to the 3PCR platform.

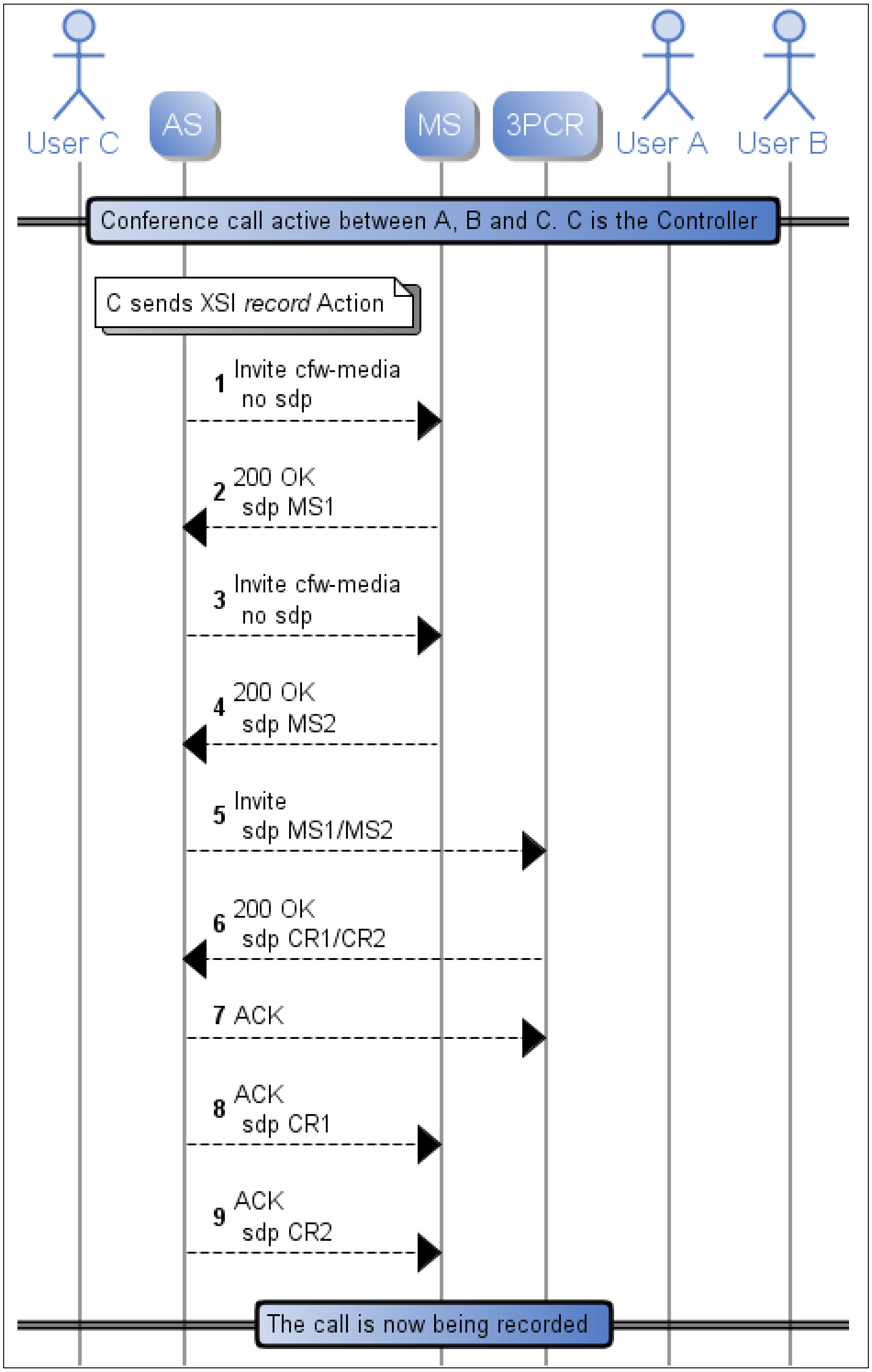

Figure 25 Start After Conference Established

10.2.7.1 INVITE to 3PCR Platform

Following is an example INVITE sent to the 3PCR platform for the start of call recording on an existing conference. The metadata contains all three parties in the conference call and there are only two media streams. The call recording is inserted between the conference controller and the conference bridge.

```
INVITE sip:3PCR@recorder.broadsoft.com:5060;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26ff4f26931B5DED
From: <sip:as1.broadsoft.com>;tag=B6CB9EB1-8AE7B5DC
To: sip:3PCR@recorder.broadsoft.com
Call-ID:BW15331149901041111215905@10.16.134.17
CSeq:157450406 INVITE
Contact:<sip:as1.broadsoft.com:5060>;src
Supported:100rel
Recv-Info:
Allow:ACK,BYE,CANCEL,INFO,INVITE,OPTIONS,PRACK,REFER,NOTIFY,UPDATE
Accept:application/media_control+xml,application/sdp,multipart/mixed
Max-Forwards:10
Content-Type: multipart/mixed;boundary=dfe2341adf13412fqdfadq
Content-Length: ...

--dfe2341adf13412fqdfadq
Content-Type: application/sdp

v=0
o=BroadWorks 17 1 IN IP4 10.16.134.101
s=-
c=IN IP4 10.16.134.101
t=0 0
m=audio 2262 RTP/AVP 8
a=rtpmap:8 PCMA/8000
a=sendonly
a=label:1
m=audio 15096 RTP/AVP 0 18 9
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=label:2


--dfe2341adf13412fqdfadq
Content-Type: application/rs-metadata+xml
Content-Disposition: recording-session

<?xml version="1.0" encoding="UTF-8"?>
<recording_metadata xmlns="urn:ietf:params:xml:ns:siprec">
                            <dataMode>complete</dataMode>
                            <recording id="urn:uuid:e33c1117-a610-4f8e-
9b3f-ce220c38c900">
                                <requestor>SRC</requestor>
                                <type>selective</type>
                            </recording>
                            <group id="urn:uuid:6e14b6e7-62ee-4421-a96d-
cd9211d36d14" recording="urn:uuid:e33c1117-a610-4f8e-9b3f-ce220c38c900">

                            <initiator>sip:north01@rtx.broadsoft.com</ini
tiator>
                            </group>
                            <session id="urn:uuid:7ca62747-e371-4ca8-
8a50-d880ad03d86a" group="urn:uuid:6e14b6e7-62ee-4421-a96d-cd9211d36d14">
```

```
                              <start-time>2011-08-10T13:32:52-
0500</start-time>
                           </session>
                           <participant id="urn:uuid:08364f25-3712-4c2f-
ba34-a609ddc5dce1" session="urn:uuid:7ca62747-e371-4ca8-8a50-
d880ad03d86a">
                              <aor>sip:north01@rtx.broadsoft.com</aor>
                              <send>
                                     <id>urn:uuid:92a95438-1fcf-4798-
99dc-e0831a19c212</id>
                              </send>
                           </participant>
                           <participant id="urn:uuid:74fa24e0-8a31-4a56-
8df4-c3c096f50f04" session="urn:uuid:7ca62747-e371-4ca8-8a50-
d880ad03d86a">
                              <aor>sip:506@rtx.broadsoft.com</aor>
                              <send>
                                     <id>urn:uuid:d72f9ad5-6d77-4a30-
b150-73c62a79d9c6</id>
                              </send>
                           </participant>
                           <participant id="urn:uuid:25165fb3-04a1-41c1-
a2cc-f13de19fafea" session="urn:uuid:7ca62747-e371-4ca8-8a50-
d880ad03d86a">
                              <aor>north00@rtx.broadsoft.com</aor>
                              <send>
                                     <id>urn:uuid:d72f9ad5-6d77-4a30-
b150-73c62a79d9c6</id>
                              </send>
                           </participant>
                           <stream id="urn:uuid:92a95438-1fcf-4798-99dc-
e0831a19c212" session="urn:uuid:7ca62747-e371-4ca8-8a50-d880ad03d86a">
                              <label>1</label>
                              <mode>separate</mode>
                           </stream>
                           <stream id="urn:uuid:d72f9ad5-6d77-4a30-b150-
73c62a79d9c6" session="urn:uuid:7ca62747-e371-4ca8-8a50-d880ad03d86a">
                              <label>2</label>
                              <mode>separate</mode>
                           </stream>
                           <extensiondata id="urn:uuid:c18e8ffc-50d7-
4131-a72b-77aa052051c5" parent="urn:uuid:7ca62747-e371-4ca8-8a50-
d880ad03d86a">
                              <broadWorksRecordingMetadata
xmlns="http://schema.broadsoft.com/broadworksCallRecording">
                                     <extTrackingID>5:1</extTrackingID>

                           <serviceProviderID>TxASDev</serviceProviderID
>
                                     <groupID>North_as90</groupID>

                           <userID>north01@rtx.broadsoft.com</userID>
                                     <callID>callhalf-111:0</callID>
                                     <callType>
                                            <origCall>

                           <callingPartyNumber>sip:+19726990601@rtx.broa
dsoft.com</callingPartyNumber>

                           <calledPartyNumber>sip:+19726990506@rtx.broad
soft.com</calledPartyNumber>
```

```
                              &lt;dialedDigits&gt;sip:506@rtx.broadsoft.com</dial
edDigits>
                                              &lt;/origCall&gt;
                                       &lt;/callType&gt;
                                       &lt;recordingType&gt;on&lt;/recordingType&gt;
                                 &lt;/broadWorksRecordingMetadata&gt;
                              &lt;/extensiondata&gt;
&lt;/recording_metadata&gt;--dfe2341adf13412fqdfadq—
```

10.2.8 Conference Call; Recording One Leg of Three-Way Conference

The following call flows show a situation in which a user is recording a two-party call who then decides to add another party to the call and initiate a conference call. Prior to the start of the call flow, User A is on a call with User B and User A is recording the call. User A has put User B on hold and sets up the call for User C. The call flow starts with User A initiating the conference between Users A, B, and C. It shows how the INVITE is sent after the conference is established and it highlights that there is only a single recording.

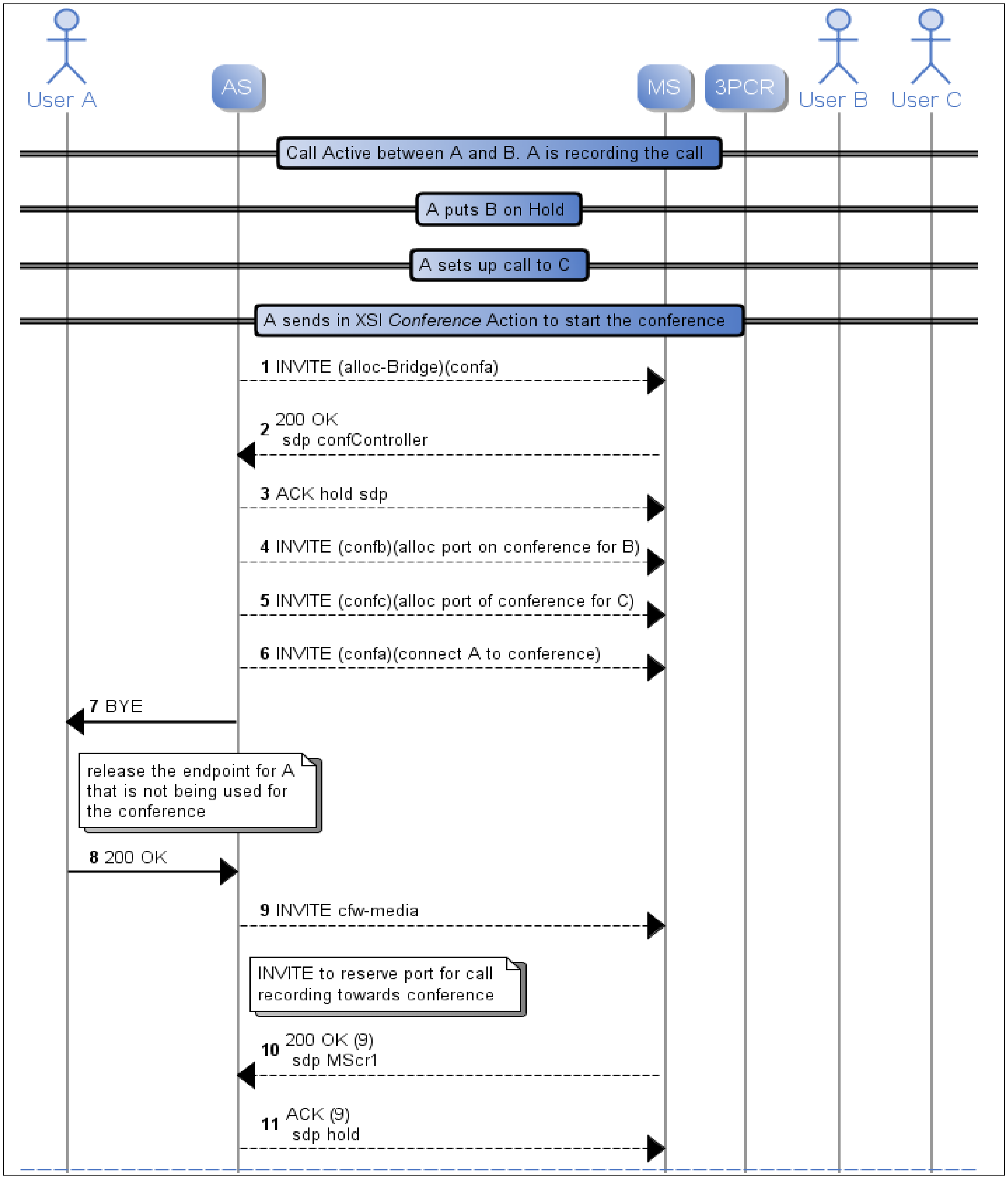

Figure 26 Conference Setup (a)

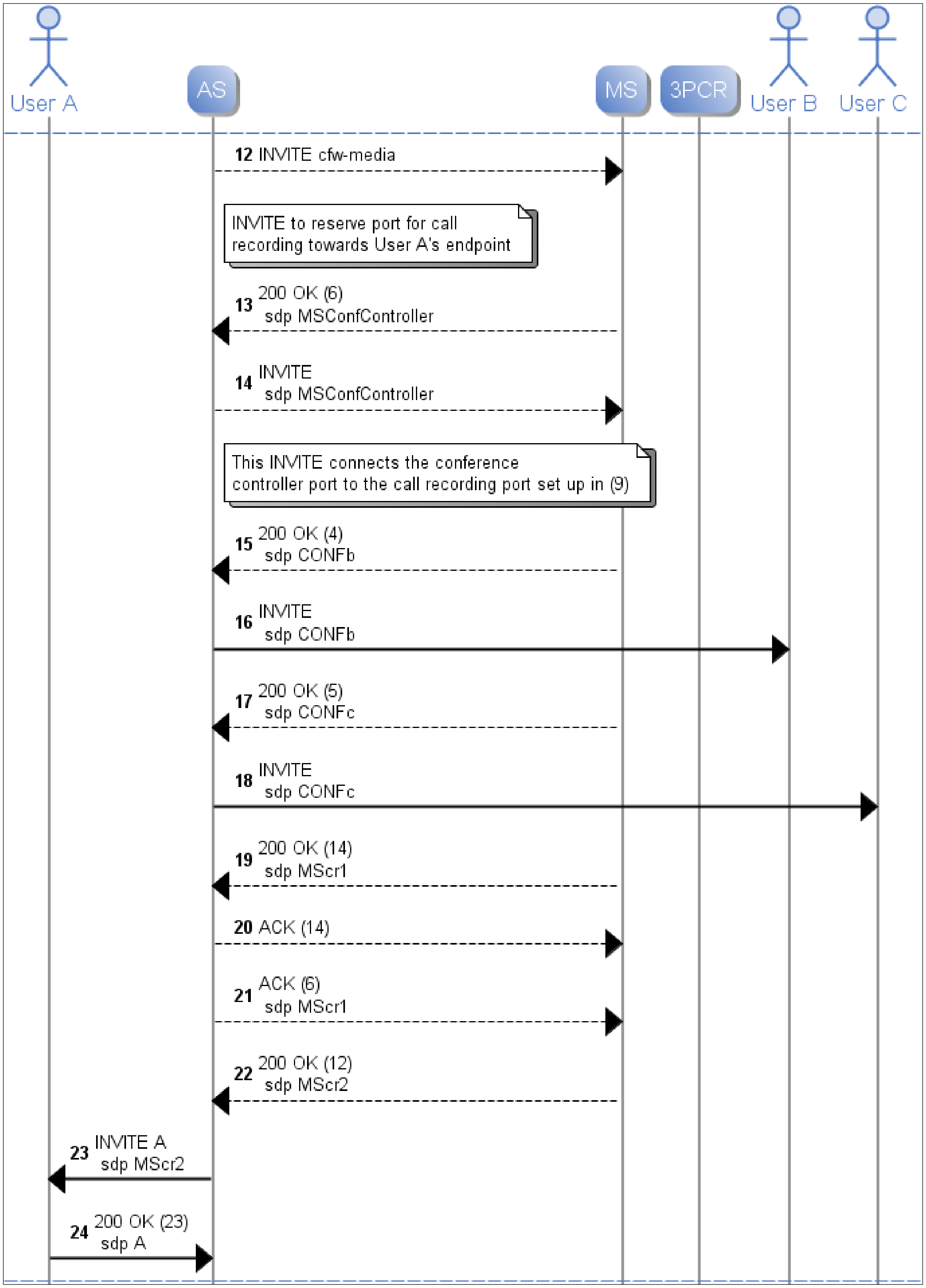

Figure 27 Conference Setup (b)

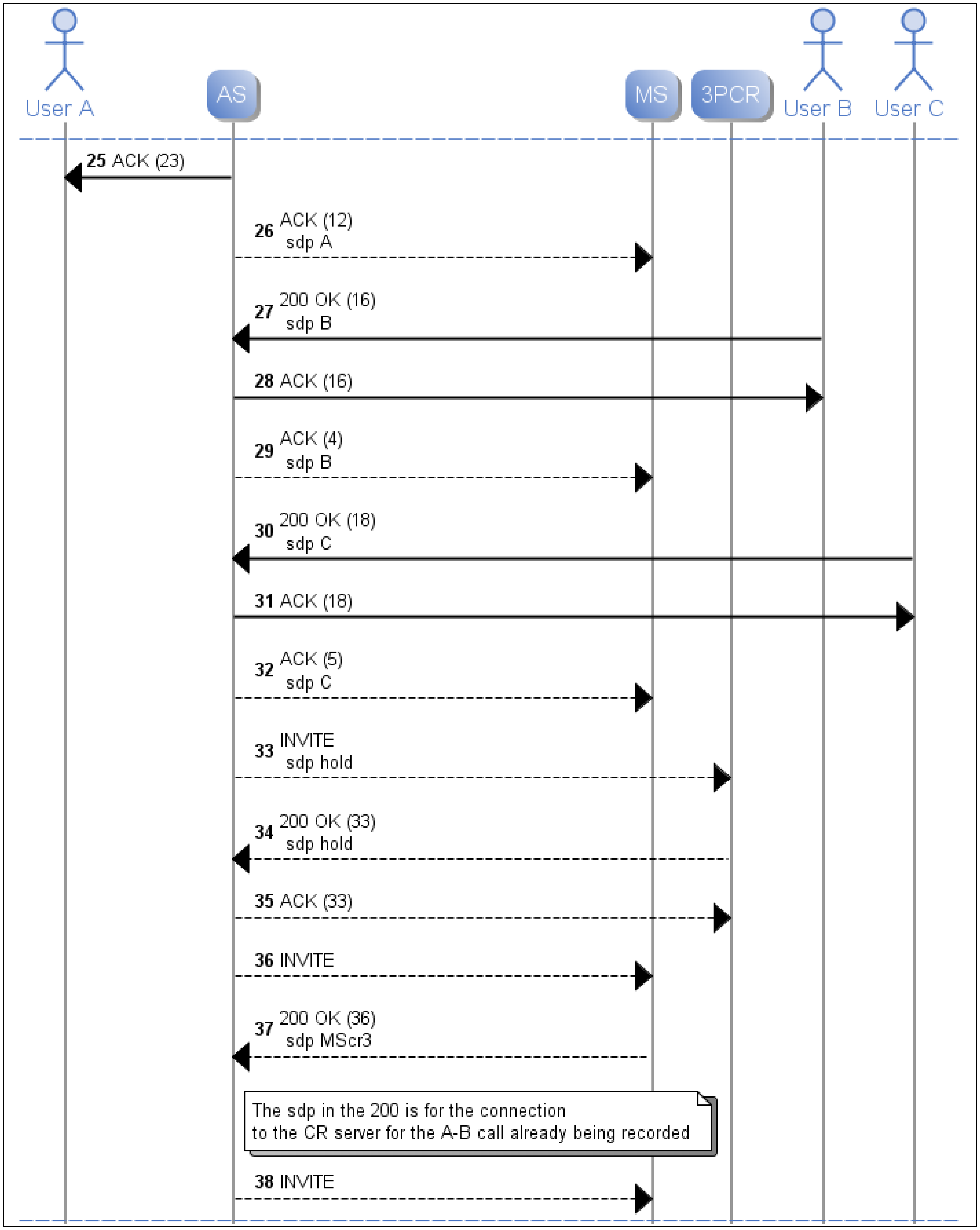

Figure 28 Conference Setup (c)

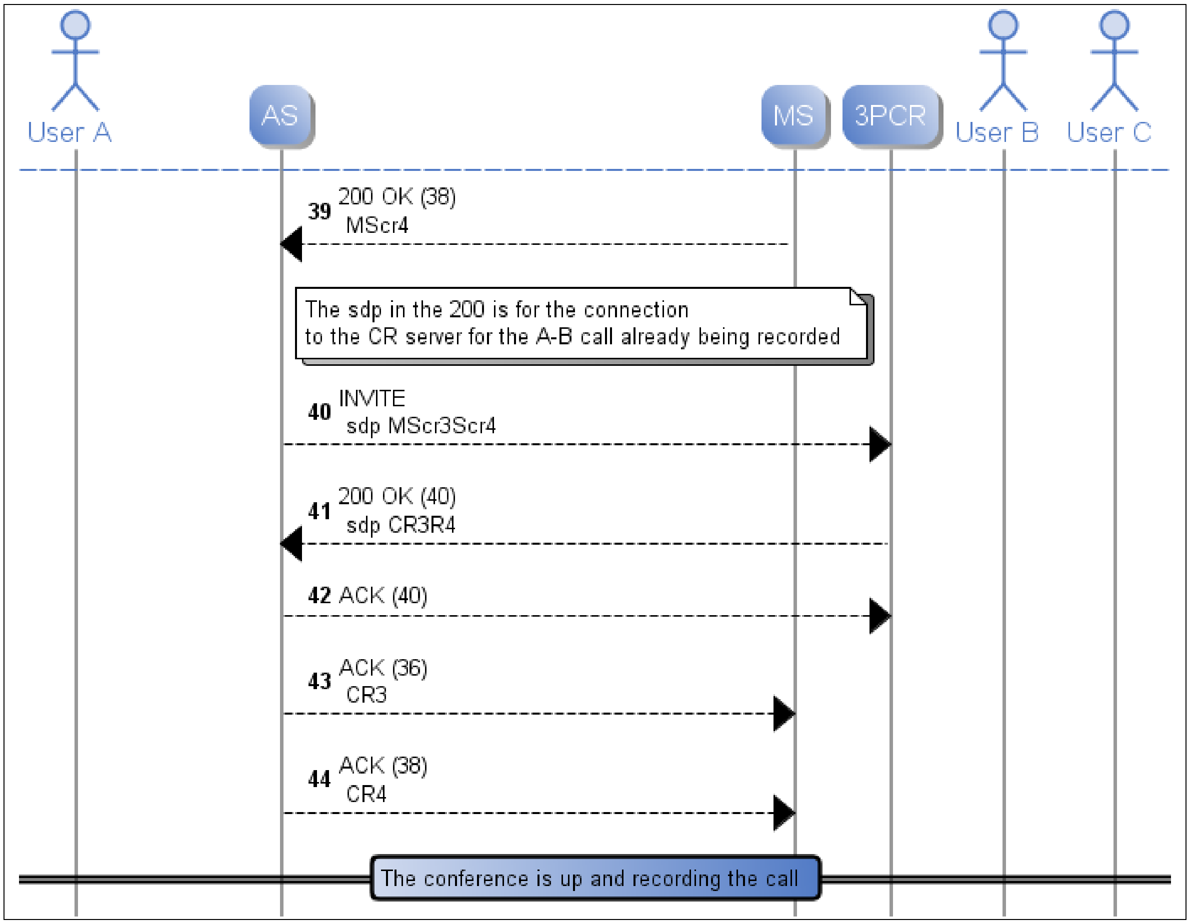

Figure 29 Conference Setup (d)

10.2.8.1 INIVTE to Put 3PCR Platform On Hold

As part of setting up the conference, the recording is put on hold temporarily. This is done by sending an INVITE with the media SDP on hold. At this point, there has been no change to the participants; however, there is no metadata sent in the INVITE.

```
INVITE sip:10.16.134.17:5070;transport=UDP SIP/2.0
Via:SIP/2.0/UDP 10.16.134.17;branch=z9hG4bKBroadWorks.1lp0lgs-
10.16.134.17V5070-0-153184596-833240841-1361811002020-
From:"User00
North"<sip:9726990600@orton.rtx.broadsoft.com;user=phone>;tag=833240841-
1361811002020-
To:<sip:10.16.134.17:5070>;tag=1
Call-ID:BW105002020250213-1185586707@10.16.134.17
CSeq:153184596 INVITE
Route:<sip:10.16.134.17:5070;lr>
Contact:<sip:10.16.134.17:5060>;src
Allow:ACK,BYE,CANCEL,INFO,INVITE,OPTIONS,PRACK,REFER,NOTIFY
Supported:
Accept:application/media_control+xml,application/sdp,application/x-
broadworks-call-center+xml
Max-Forwards:10
Content-Type:application/sdp
Content-Length:604

v=0
```

```
o=BroadWorks 6954 1 IN IP4 10.16.120.22
s=-
c=IN IP4 0.0.0.0
t=0 0
m=audio 21330 RTP/AVP 0 8 18 96 9 2 101
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:96 AMR/8000
a=rtpmap:9 G722/8000
a=rtpmap:2 G726-32/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=ptime:20
a=label:1
a=inactive
m=audio 21288 RTP/AVP 0 8 18 96 9 2 101
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:96 AMR/8000
a=rtpmap:9 G722/8000
a=rtpmap:2 G726-32/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=ptime:20
a=label:2
a=inactive
```

10.2.8.2 INVITE Message for 3PCR Platform

Following is an example of the second INVITE sent to the 3PCR platform when the conference is established. Note that User C is added as a participant.

```
INVITE sip:10.16.134.17:5070;transport=UDP SIP/2.0
Via:SIP/2.0/UDP 10.16.134.17;branch=z9hG4bKBroadWorks.1lp0lgs-
10.16.134.17V5070-0-153184597-833240841-1361811002020-
From:"User00
North"<sip:9726990600@orton.rtx.broadsoft.com;user=phone>;tag=833240841-
1361811002020-
To:<sip:10.16.134.17:5070>;tag=1
Call-ID:BW105002020250213-1185586707@10.16.134.17
CSeq:153184597 INVITE
Route:<sip:10.16.134.17:5070;lr>
Contact:<sip:10.16.134.17:5060>;src
Allow:ACK,BYE,CANCEL,INFO,INVITE,OPTIONS,PRACK,REFER,NOTIFY
Supported:
Accept:application/media_control+xml,application/sdp,application/x-
broadworks-call-center+xml
Max-Forwards:10
Content-Type:multipart/mixed;boundary=UniqueBroadWorksBoundary
Content-Length:3283
MIME-Version:1.0


--UniqueBroadWorksBoundary
Content-Type:application/rs-metadata+xml
Content-Disposition:recording-session
Content-Length:2426

<?xml version="1.0" encoding="UTF-8"?>
```

```
<recording_metadata xmlns="urn:ietf:params:xml:ns:siprec">
  <dataMode>complete</dataMode>
  <recording id="urn:uuid:8e0318e8-f0d3-41a2-a5b6-749d0f6a3bf3">
    <requestor>SRC</requestor>
    <type>selective</type>
  </recording>
  <group id="urn:uuid:ba9447a3-eef1-47e3-bad5-6fcda5486620"
recording="urn:uuid:8e0318e8-f0d3-41a2-a5b6-749d0f6a3bf3">
    <initiator>sip:north00@orton.rtx.broadsoft.com</initiator>
  </group>
  <session id="urn:uuid:a57e8b1f-81ba-40de-8025-742e466b5959"
group="urn:uuid:ba9447a3-eef1-47e3-bad5-6fcda5486620">
    <start-time>2013-02-25T10:50:02-06:00</start-time>
  </session>
  <participant id="urn:uuid:b3537ce4-ef7f-4686-801a-b8a748dfcd52"
session="urn:uuid:a57e8b1f-81ba-40de-8025-742e466b5959">
    <aor>sip:north00@orton.rtx.broadsoft.com</aor>
    <send>
      <id>urn:uuid:d23e4862-6983-45e9-b479-071d80093947</id>
    </send>
  </participant>
  <participant id="urn:uuid:c1258cfc-b2ac-4fa9-8082-a66bf98157ac"
session="urn:uuid:a57e8b1f-81ba-40de-8025-742e466b5959">
    <aor>sip:506@orton.rtx.broadsoft.com</aor>
    <send>
      <id>urn:uuid:ec060bd6-599f-42dc-bf55-0f3636f698d3</id>
    </send>
  </participant>
  <participant id="urn:uuid:ba0b55dd-f791-4fca-b34b-a0ea4d387c03"
session="urn:uuid:a57e8b1f-81ba-40de-8025-742e466b5959">
    <aor>north01@orton.rtx.broadsoft.com</aor>
    <send>
      <id>urn:uuid:ec060bd6-599f-42dc-bf55-0f3636f698d3</id>
    </send>
  </participant>
  <stream id="urn:uuid:d23e4862-6983-45e9-b479-071d80093947"
session="urn:uuid:a57e8b1f-81ba-40de-8025-742e466b5959">
    <label>1</label>
    <mode>separate</mode>
  </stream>
  <stream id="urn:uuid:ec060bd6-599f-42dc-bf55-0f3636f698d3"
session="urn:uuid:a57e8b1f-81ba-40de-8025-742e466b5959">
    <label>2</label>
    <mode>separate</mode>
  </stream>
  <extensiondata id="urn:uuid:178210bd-bb1a-4d13-adfe-3dacffd06e46"
parent="urn:uuid:a57e8b1f-81ba-40de-8025-742e466b5959">
    <broadWorksRecordingMetadata
xmlns="http://schema.broadsoft.com/broadworksCallRecording"
version="2.0">
      <extTrackingID>15:1</extTrackingID>
      <serviceProviderID>TxASDev</serviceProviderID>
      <groupID>North_as90</groupID>
      <userID>north00@orton.rtx.broadsoft.com</userID>
      <callID>callhalf-27527:0</callID>
      <callType>
        <origCall>

<callingPartyNumber>sip:+19726990600@orton.rtx.broadsoft.com</callingPart
yNumber>

<calledPartyNumber>sip:506@orton.rtx.broadsoft.com</calledPartyNumber>
```

```
          <dialedDigits>sip:506@orton.rtx.broadsoft.com</dialedDigits>
        </origCall>
      </callType>
      <recordingType>on</recordingType>
    </broadWorksRecordingMetadata>
  </extensiondata>
</recording_metadata>
--UniqueBroadWorksBoundary
Content-Type:application/sdp
Content-Length:609

v=0
o=BroadWorks 6971 1 IN IP4 10.16.120.22
s=-
c=IN IP4 10.16.120.22
t=0 0
m=audio 21364 RTP/AVP 0 101 8 18 96 9 2
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:96 AMR/8000
a=rtpmap:9 G722/8000
a=rtpmap:2 G726-32/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=ptime:20
a=sendonly
a=label:1
m=audio 21334 RTP/AVP 0 101 8 18 96 9 2
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:96 AMR/8000
a=rtpmap:9 G722/8000
a=rtpmap:2 G726-32/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=ptime:20
a=sendonly
a=label:2

--UniqueBroadWorksBoundary--
```

10.2.9 Add Call Being Recorded to Active Conference

The following call flow starts with a call from User A to User D, which is being recorded. User A then adds this call to an existing conference between Users A, B, and C. The recording for the User A to User D call is held while the recording is added to the conference. Once this is done, the call recording is updated with new metadata, which includes adding Users B and C to the recording.

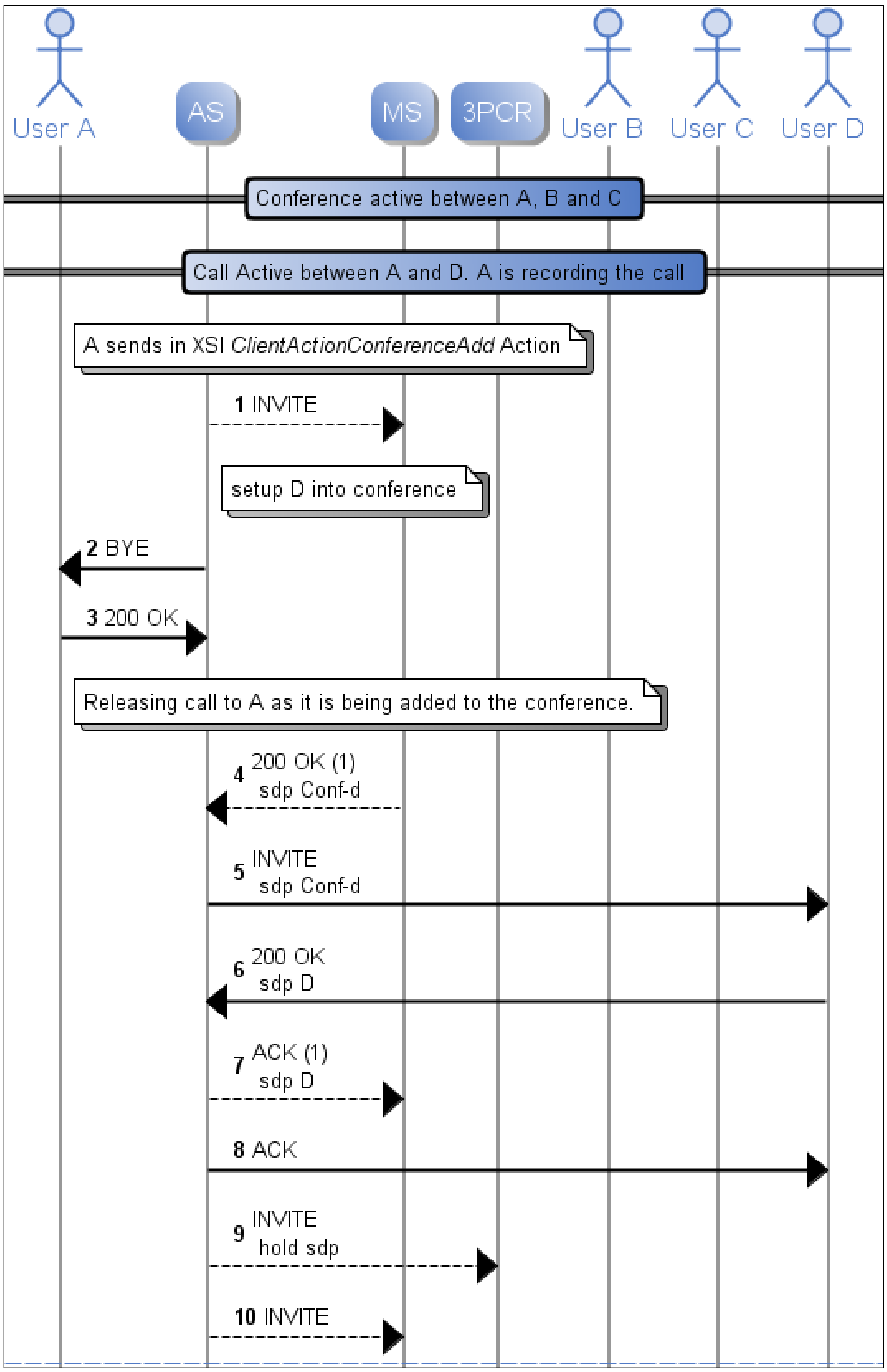

Figure 30 Recorded Call Being Added to Conference (a)

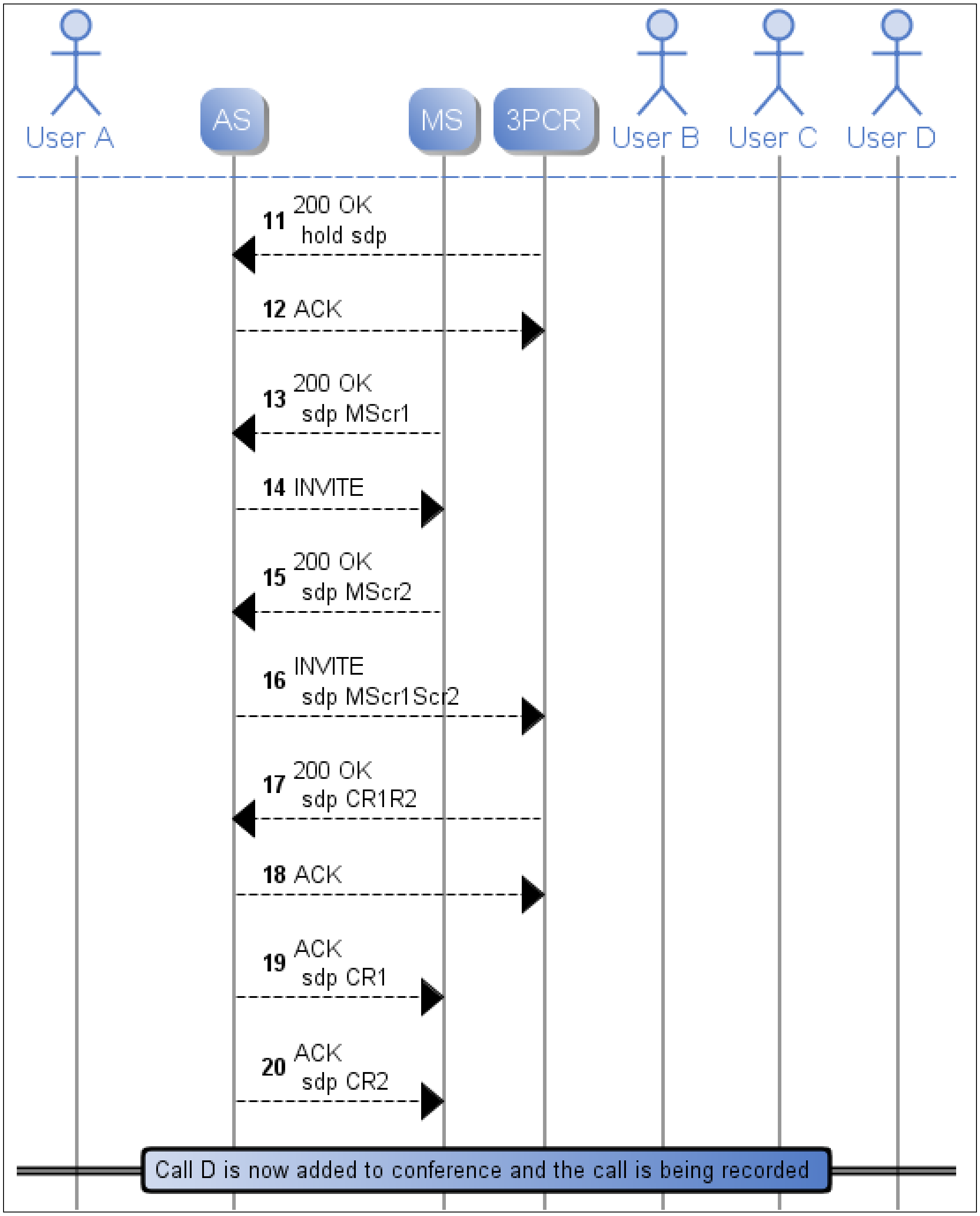

Figure 31 Recorded Call Being Added to Conference (b)

10.2.9.1 INVITE Message for 3PCR Platform

Following is an example of the second INVITE sent to the 3PCR platform when User D is added to the conference. Note that Users B and C are added to the metadata as they are now also being recorded.

```
INVITE sip:10.16.134.17:5070;transport=UDP SIP/2.0
Via:SIP/2.0/UDP 10.16.134.17;branch=z9hG4bKBroadWorks.1lp0lgs-
10.16.134.17V5070-0-779834172-1850244762-1352326882930-
```

```
From:"Scott
North"<sip:9726990600@rtx.broadsoft.com;user=phone>;tag=1850244762-
1352326882930-
To:<sip:10.16.134.17:5070>;tag=4
Call-ID:BW1621229300711121414075369@10.16.134.17
CSeq:779834172 INVITE
Route:<sip:10.16.134.17:5070;lr>
Contact:<sip:10.16.134.17:UserB0>;src
Max-Forwards:10
Content-Type:multipart/mixed;boundary=UniqueBroadWorksBoundary
Content-Length: ...
MIME-Version:1.0


--UniqueBroadWorksBoundary
Content-Type:application/rs-metadata+xml
Content-Disposition:recording-session
Content-Length: ...

<?xml version="1.0" encoding="UTF-8"?>
<recording_metadata xmlns="urn:ietf:params:xml:ns:siprec">
                            <dataMode>complete</dataMode>
                            <recording id="urn:uuid:5c1f48e9-e796-4140-
9b60-0fde3e066c97">
                                <requestor>SRC</requestor>
                                <type>selective</type>
                            </recording>
                            <group id="urn:uuid:ef6ad558-3468-45d4-89d6-
53a892d7872b" recording="urn:uuid:5c1f48e9-e796-4140-9b60-0fde3e066c97">

                            <initiator>sip:UserA@rtx.broadsoft.com</initi
ator>
                            </group>
                            <session id="urn:uuid:9fe0bfb3-e3e8-4b5c-
b2cf-ec42dc6230e5" group="urn:uuid:ef6ad558-3468-45d4-89d6-53a892d7872b">
                                <start-time>2012-11-07T16:21:22-
06:00</start-time>
                            </session>
                            <participant id="urn:uuid:2c60a7ef-b354-488a-
99bc-a1e2408362c5" session="urn:uuid:9fe0bfb3-e3e8-4b5c-b2cf-
ec42dc6230e5">
                                <aor>sip:UserA@rtx.broadsoft.com</aor>
                                <send>
                                       <id>urn:uuid: c90b67bd-82bd-4fee-
8835-d70d6e1eb809</id>
                                </send>
                            </participant>
    <participant id="urn:uuid:cbc24326-8e15-494c-b0e4-18da69808976"
session="urn:uuid:9fe0bfb3-e3e8-4b5c-b2cf-ec42dc6230e5">
                                <aor>sip:UserD@rtx.broadsoft.com</aor>
                                <send>
                                       <id>urn:uuid: c90b67bd-82bd-4fee-
8835-d70d6e1eb809</id>
                                </send>
                            </participant>
                            <participant id="urn:uuid:450423fd-a5bc-4a6e-
a73b-1cd1ef390580" session="urn:uuid:9fe0bfb3-e3e8-4b5c-b2cf-
ec42dc6230e5">
                                <aor>sip:UserB@rtx.broadsoft.com</aor>
                                <send>
                                       <id>urn:uuid: c90b67bd-82bd-4fee-
8835-d70d6e1eb809</id>
```

```
                              </send>
                           </participant>
                           <participant id="urn:uuid:cbc24326-8e15-494c-
b0e4-18da6980c437" session="urn:uuid:9fe0bfb3-e3e8-4b5c-b2cf-
ec42dc6230e5">
                              <aor>sip:UserC@rtx.broadsoft.com</aor>
                              <send>
                                     <id>urn:uuid: c90b67bd-82bd-4fee-
8835-d70d6e1eb809</id>
                              </send>
                           </participant>
                           <stream id="urn:uuid:c90b67bd-82bd-4fee-8835-
d70d6e1eb809" session="urn:uuid:9fe0bfb3-e3e8-4b5c-b2cf-ec42dc6230e5">
                              <label>1</label>
                              <mode>separate</mode>
                           </stream>
                           <extensiondata id="urn:uuid:6a4272a6-d92c-
46ea-8ff9-3f6471484195" parent="urn:uuid:9fe0bfb3-e3e8-4b5c-b2cf-
ec42dc6230e5">
                              <broadWorksRecordingMetadata
xmlns="http://schema.broadsoft.com/broadworksCallRecording"
version="2.0">
                                     <extTrackingID>8:1</extTrackingID>

                           <serviceProviderID>TxASDev</serviceProviderID
>
                                     <groupID>North_as90</groupID>

                           <userID>UserA@rtx.broadsoft.com</userID>
                                     <callID>callhalf-121:1</callID>
                                     <callType>
                                            <origCall>

                           <callingPartyNumber>sip:+19726990600@rtx.broa
dsoft.com</callingPartyNumber>

                           <calledPartyNumber>sip:506@rtx.broadsoft.com<
/calledPartyNumber>

                           <dialedDigits>sip:506@rtx.broadsoft.com</dial
edDigits>
                                            </origCall>
                                     </callType>
                                     <recordingType>on</recordingType>
                              </broadWorksRecordingMetadata>
                           </extensiondata>
</recording_metadata>
--UniqueBroadWorksBoundary
Content-Type:application/sdp
Content-Length: ...

v=0
o=BroadWorks 156 1 IN IP4 10.16.120.22
s=-
c=IN IP4 10.16.120.22
t=0 0
m=audio 12382 RTP/AVP 0 8 18 96 9 2 101
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:96 AMR/8000
a=rtpmap:9 G722/8000
```

```
a=rtpmap:2 G726-32/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=ptime:20
a=sendonly
a=label:1

--UniqueBroadWorksBoundary--
```

10.2.10 Add Call Being Recorded to Conference Being Recorded

This call flow is identical to the flow shown in section *10.2.9 Add Call Being Recorded to Active Conference*. The call recording already active on the conference is not affected since it is already active. The only changes are to the recording for the call being added to the conference.

10.2.11 Always with Pause/Resume Support

10.2.11.1 Pause

The following call flow shows the call recording being paused. In this situation, the user is initiating the pause using the FAC instead of the Xtended Services Interface event. The messages sent to the 3PCR platform and the Media Server are similar to those shown in section *10.2 Start, Stop, Pause, and Resume*. This call flow picks up after the call flow shown in *Figure 16*. However the recording type sent in the metadata is *on*.

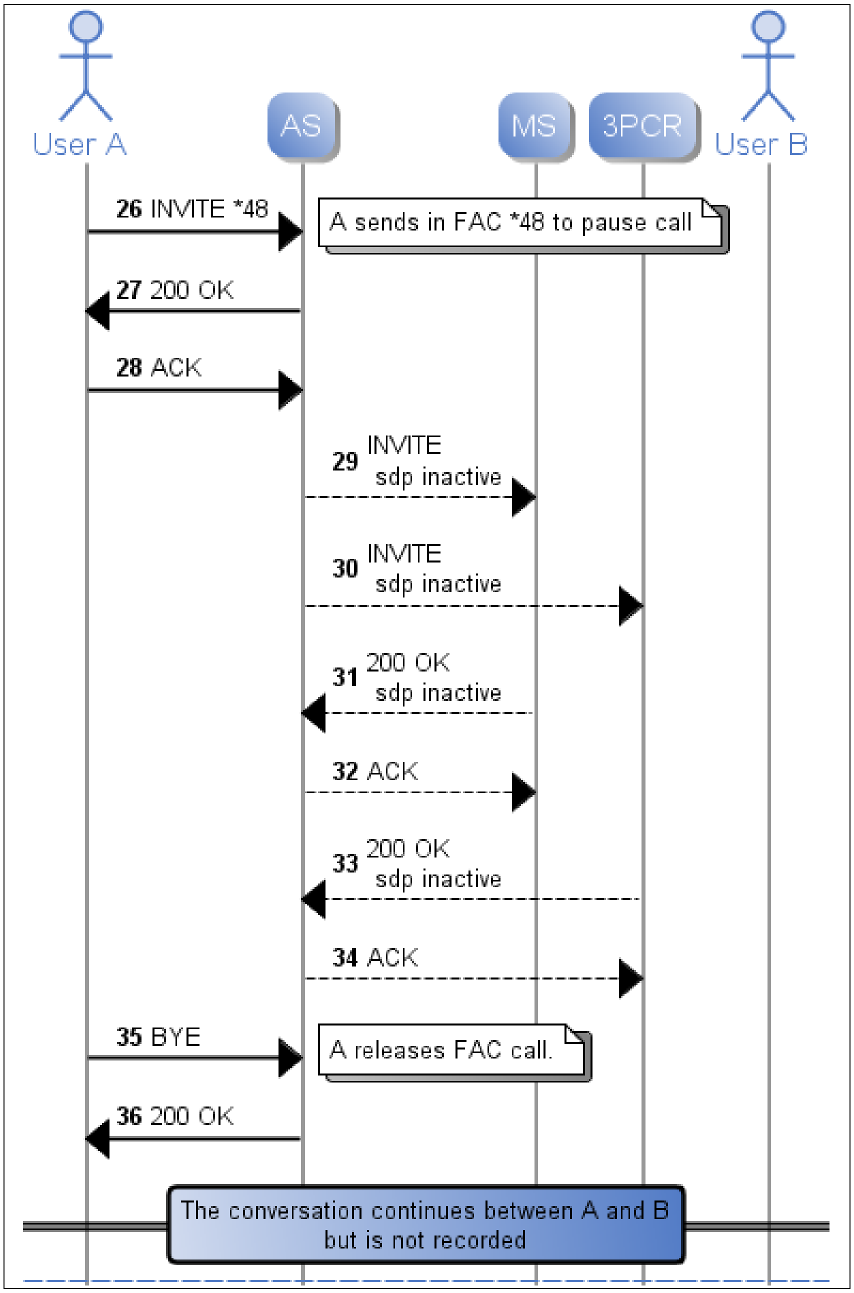

Figure 32 Always Pause Recording

10.2.11.2 Resume

The following call flow shows the call recording being paused. In this situation, the user is initiating the pause using the FAC instead of the Xtended Services Interface event. The messages sent to the 3PCR platform and the Media Server are similar to those shown in section *10.2 Start, Stop, Pause, and Resume*.

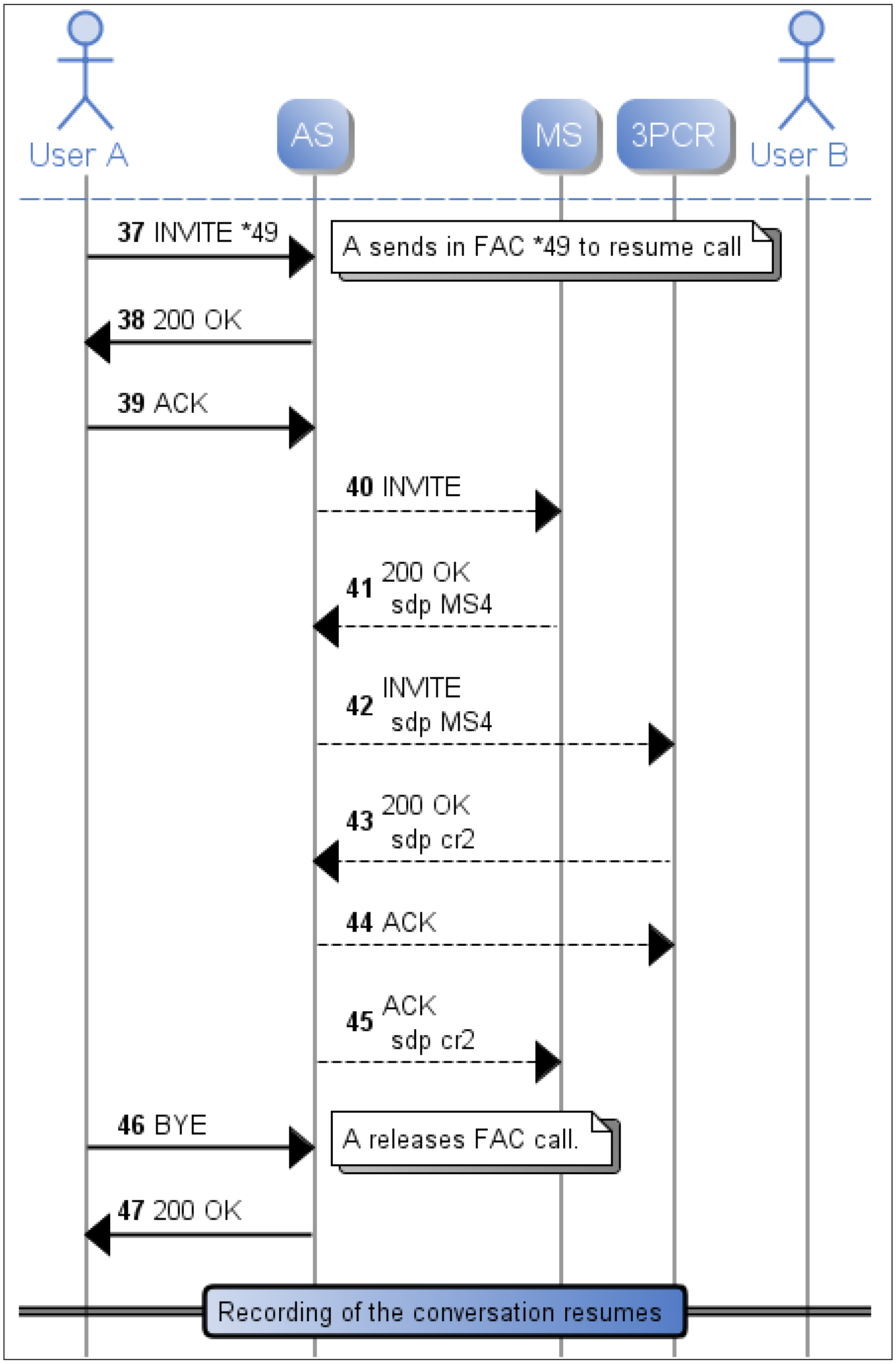

Figure 33 Always Resume Recording

10.3 Video Call Flows

This section shows some of the basic call flows that involve video. In the call flows, the 100 trying messages are skipped for the sake of brevity. For this same reason, only headers that are important to the call flows are shown. This means some mandatory SIP headers are missing. The call flows highlight the headers that are important for the interface to the Call Recording platform.

These call flows all have a generic pattern that involve the following steps:

- The end users negotiate end-to-end and establish the call with audio and video successfully.
- The Cisco BroadWorks Telephony Application Server negotiates with 3PCR with a filtered list of codecs based on the offer/answer of the end users and the Media Server.
- The Cisco BroadWorks Telephony Application Server then reconnects the end users through the Media Server.
- If there are any changes in the call topology or SDP, the Cisco BroadWorks Telephony Application Server may either send updates or renegotiate with 3PCR.

10.3.1 Messaging Between Cisco BroadWorks Telephony Application Server and 3PCR Platform to Show Metadata

The following messages show the format of the SDP and the metadata sent to the 3PCR platform for calls with audio and video. Note that the changes due to video are highlighted in bold.

Application Server to 3PCR platform

```
INVITE sip:3PCR@recorder.broadsoft.com:5060;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26f33f26931B5DED
From: <sip:as1.broadsoft.com>;tag=B6CB9EB1-8AE7B75C
To: sip:3PCR@recorder.broadsoft.com
Call-ID:BW153311499010488569905@10.16.134.17
CSeq:25 INVITE
Contact:<sip:as1.broadsoft.com:5060>;src
Supported:100rel
Recv-Info:
Allow:ACK,BYE,CANCEL,INFO,INVITE,OPTIONS,PRACK,REFER,NOTIFY,UPDATE
Accept:application/media_control+xml,application/sdp,multipart/mixed
Max-Forwards:10
Content-Type: multipart/mixed;boundary=deew341adf13412ferwadq
Content-Length: ...

-- deew341adf13412ferwadq
Content-Type: application/sdp

v=0
o=BroadWorks 783 1 IN IP4 10.16.120.22
s=-
c=IN IP4 10.16.120.22
t=0 0
m=audio 25980 RTP/AVP 0 8 18 96 9 2 101
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:96 AMR/8000
a=rtpmap:9 G722/8000
a=rtpmap:2 G726-32/8000
a=rtpmap:101 telephone-event/8000
```

```
a=fmtp:101 0-15
a=ptime:20
a=sendonly
a=label:1
m=video 25978 RTP/AVP 104
b=AS:512
a=rtpmap:104 H264/90000
a=fmtp:104 profile-level-id=42E00D; packetization-mode=0
a=sendOnly
a=label:3
m=audio 26042 RTP/AVP 0 8 18 96 9 2 101
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:96 AMR/8000
a=rtpmap:9 G722/8000
a=rtpmap:2 G726-32/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=ptime:20
a=sendonly
a=label:2
m=video 26040 RTP/AVP 104 b=AS:512
a=rtpmap:104 H264/90000
a=fmtp:104 profile-level-id=42E00D; packetization-mode=0
a=sendOnly
a=label:4


-- deew341adf13412ferwadq
Content-Type: application/rs-metadata+xml
Content-Disposition: recording-session

<?xml version="1.0" encoding="UTF-8"?>
<recording-metadata xmlns="urn:ietf:params:xml:ns:recording"
id="urn:uuid:34512345-6743-6248-9043897645ab">
  <dataMode>complete</dataMode>
  <recording id="urn:uuid:894134ab-9800-7844-4456-789451125647">
    <requestor>src</requestor>
  </recording>
  <group id="urn:uuid:abc12785-4788-6654-5455-45def4522375"
         recording="urn:uuid:894134ab-9800-7844-4456-789451125647"/>
  <session id="urn:uuid:78554655-7844-5564-4568-ef4566246875"
           group="urn:uuid:abc12785-4788-6654-5455-45def4522375">
  </session>
<participant id="urn:uuid:e0471d38-e2eb-46a2-b486-47cfeda8a45a"
session="urn:uuid: 78554655-7844-5564-4568-ef4566246875">
    <aor> north02@rtx.broadsoft.com </aor>
    <send>
           <id>urn:uuid: 94dwif31-9887-341d-12id-789945621002</id>
    </send>
     <send>
           <id>urn:uuid: 94dwif31-9887-341d-12id-789945621003</id>
    </send>
</participant>
<participant id="urn:uuid:e0471d38-e2eb-46a2-b486-47cfeda8a45a"
session="urn:uuid: 78554655-7844-5564-4568-ef4566246875">
    <aor> north03@rtx.broadsoft.com </aor>
    <send>
           <id>urn:uuid: 34123561-7789-341d-12id-78edcaf78945</id>
    </send>
     <send>
```

```
          <id>urn:uuid: 34123561-7789-341d-12id-78edcaf78946</id>
    </send>
</participant>

  <stream id="urn:uuid:94dwif31-9887-341d-12id-789945621002"
          session="urn:uuid:78554655-7844-5564-4568-ef4566246875">
    <label>1</label>
    <mode>mixed</mode>
  </stream>
  <stream id="urn:uuid:94dwif31-9887-341d-12id-789945621003"
          session="urn:uuid:78554655-7844-5564-4568-ef4566246875">
    <label>2</label>
    <mode>mixed</mode>
  </stream>
  <stream id="urn:uuid:34123561-7789-341d-12id-78edcaf78945"
          session="urn:uuid:78554655-7844-5564-4568-ef4566246875">
    <label>3</label>
    <mode>mixed</mode>
  </stream>
  <stream id="urn:uuid:34123561-7789-341d-12id-78edcaf78946"
          session="urn:uuid:78554655-7844-5564-4568-ef4566246875">
    <label>4</label>
    <mode>mixed</mode>
  </stream>

  <extensionData id="urn:uuid:ef45678456-4451-4568-7785-400554586487"
                 parent="urn:uuid:78554655-7844-5564-4568-ef4566246875">
    <broadWorksRecordingMetadata
xmlns="http://schema.broadsoft.com/broadworksCallRecording"
schemaRev="1.0"
xsi:schemaLocation="http://schema.broadsoft.com/broadworksCallRecording"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <extTrackingID>4:1</extTrackingID>
      <serviceProviderID>TxASDev</serviceProviderID>
      <groupID>North_as90</groupID>
      <userID>north02@rtx.broadsoft.com</userID>
      <callID> BW153311411129885@10.16.134.17</callID>
      <callType>
        <origCall>
          <callingPartyNumber>north03@broadsoft.com</callingPartyNumber>
          <calledPartyNumber>2146415689</calledPartyNumber>
          <dialedDigits>2145551212</dialedDigits>
        </origCall>
      </callType>
      <recordingType>demand</recordingType>
    </broadWorksRecordingMetadata>
  </extensionData>
</recording-metadata>
-- deew341adf13412ferwadq
```

3PCR Platform to Application Server

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26f33f26931B5DED
From: <sip:as1.broadsoft.com>;tag=B6CB9EB1-8AE7B75C
To: sip:3PCR@recorder.broadsoft.com;tag=45DEF34123412DF
Call-ID: BW15331149901048856990 5@10.16.134.17
CSeq:25 INVITE
Contact:<sip:recorder.broadsoft.com:5060>;srs
Recv-Info: x-broadworks-callrecording
Allow:ACK,BYE,CANCEL,INFO,INVITE,OPTIONS,PRACK,REFER,NOTIFY,UPDATE
Content-Type: application/sdp
```

```
Content-Length: 500

v=0
o=SIPP 144 0 IN IP4 10.16.134.17
s=Call Recording SDP
t=0 0
m=audio 6008 RTP/AVP 0 18 9
c=IN IP4 10.16.134.17
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=label:1
m=video 6004 RTP/AVP 104
b=AS:512
a=rtpmap:104 H264/90000
a=fmtp:104 profile-level-id=42E00D; packetization-mode=0
a=label:3
m=audio 6010 RTP/AVP 0 18 9
c=IN IP4 10.16.134.17
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=label:2
m=video 6006 RTP/AVP 104
b=AS:512
a=rtpmap:104 H264/90000
a=fmtp:104 profile-level-id=42E00D; packetization-mode=0
a=label:4
```

Application Server to 3PCR Platform

```
ACK sip:recorder.broadsoft.com SIP/2.0
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26ff4f268799455
From: <sip:as1.broadsoft.com>;tag= B6CB9EB1-8AE7B75C
To: sip:3PCR@recorder.broadsoft.com;tag=45DEF34123412DF
Call-ID: BW153311499010488569905@10.16.134.17
CSeq:25 ACK
Contact:<sip:as1.broadsoft.com:5060>;src
Content-Length: 0
```

10.3.2 Basic Video Call Recording in Dual Mode

The call flow shows a basic two-party call with video in dual mode. The call is established between Users A and B with the SDP negotiated for audio and video end to end. The Cisco BroadWorks Telephony Application Server then establishes connections for the Call Recording (CR) platform on the Media Server and reconnects the users through the Media Server. Sessions are also established on the Media Server for Users A and B.

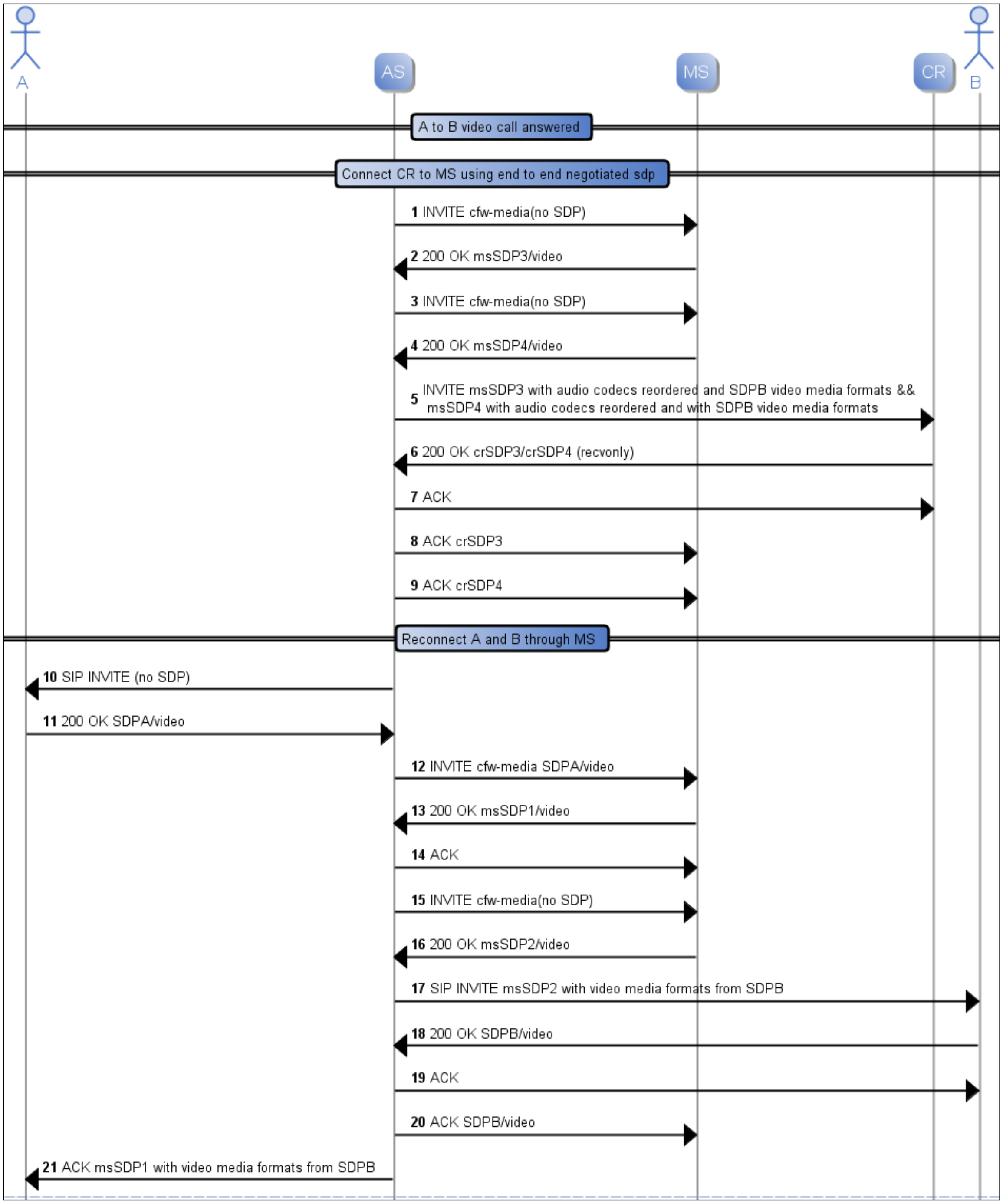

Figure 34 Basic Video Recording in Dual Mode (a)

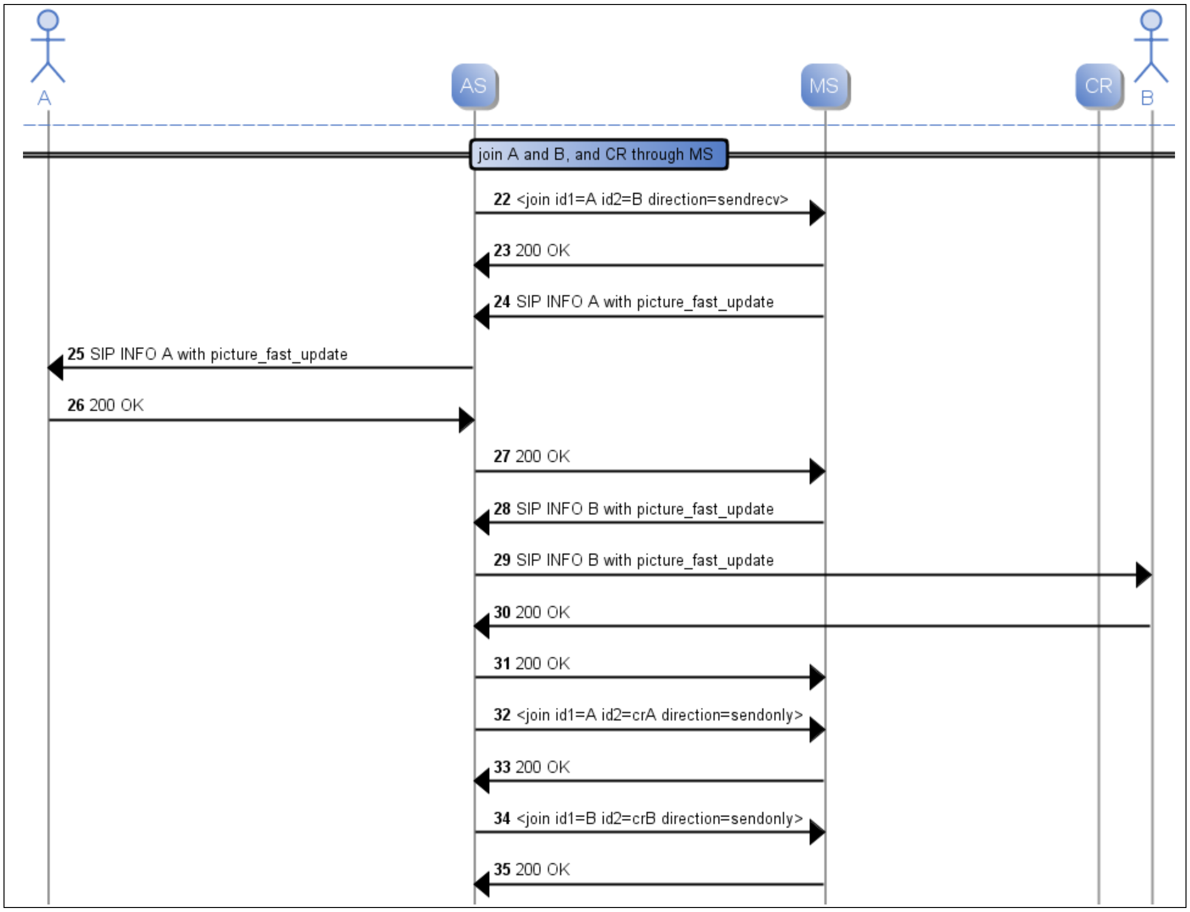

Figure 35 Basic Video Recording in Dual Mode (b)

10.3.3 Messaging to Show SDP Negotiation

Note that only the messages relevant for SDP negotiation are shown here and messages such as provisional responses, ACK without SDP, and so on are not shown (for clarity).

Initial Offer from User A

```
v=0
o=- 1360011625 1360011625 IN IP4 10.16.134.101
s=Polycom IP Phone
c=IN IP4 10.16.134.101
t=0 0
a=sendrecv
m=audio 2228 RTP/AVP 9 0 8 18 127
a=rtpmap:9 G722/8000
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=fmtp:18 annexb=no
a=rtpmap:127 telephone-event/8000
m=video 2224 RTP/AVP 109 110 34
b=AS:512
a=sendrecv
a=rtpmap:109 H264/90000
a=fmtp:109 profile-level-id=42E00C; packetization-mode=0
a=rtpmap:110 H264/90000
a=fmtp:110 profile-level-id=42800d;max-mbps=40500;max-fs=1344
```

```
a=rtpmap:34 H263/90000
a=fmtp:34 QCIF=1;SQCIF=1
```

Answer from User B to User A

```
v=0
o=- 1360011671 1360011671 IN IP4 10.16.134.100
s=Polycom IP Phone
c=IN IP4 10.16.134.100
t=0 0
a=sendrecv
m=audio 2230 RTP/AVP 0 127
a=rtpmap:0 PCMU/8000
a=rtpmap:127 telephone-event/8000
m=video 2226 RTP/AVP 109 110 34
b=AS:512
a=sendrecv
a=rtpmap:109 H264/90000
a=fmtp:109 profile-level-id=42E00D; packetization-mode=0
a=rtpmap:110 H264/90000
a=fmtp:110 profile-level-id=42800d;max-mbps=40500;max-fs=1344
a=rtpmap:34 H263/90000
a=fmtp:34 QCIF=1;SQCIF=2
```

Message 1: Application Server to Media Server

```
 INVITE cfw-media (No SDP)
```

Message 2: Answer from Media Server to Application Server

```
200 OK(MS SDP3)

v=0
o=BroadWks 14658 0 IN IP4 10.16.120.22
s=Media Server SDP
c=IN IP4 10.16.120.22
t=0 0
m=audio 25980 RTP/AVP 0 8 18 96 9 2 101
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:96 AMR/8000
a=rtpmap:9 G722/8000
a=rtpmap:2 G726-32/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=ptime:20
m=video 25978 RTP/AVP 104 105 106 107
a=rtpmap:104 H264/90000
a=fmtp:104 profile-level-id=42000C
a=rtpmap:105 H264/90000
a=ftmp:105 profile-level-id=42800a
a=rtpmap:106 H264/90000
a=fmtp:106 profile-level-id=42800c
a=rtpmap:107 H263/90000
a=fmtp:107 QCIF=1;SQCIF=1
```

Message 3: Application Server to Media Server

```
INVITE cfw-media (No SDP)
```

Message 4: 200 OK (Media Server SDP4)

```
v=0
o=BroadWks 14659 0 IN IP4 10.16.120.22
s=Media Server SDP
c=IN IP4 10.16.120.22
t=0 0
m=audio 26042 RTP/AVP 0 8 18 96 9 2 101
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:96 AMR/8000
a=rtpmap:9 G722/8000
a=rtpmap:2 G726-32/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=ptime:20
m=video 26040 RTP/AVP 104 105 106 107
a=rtpmap:104 H264/90000
a=fmtp:104 profile-level-id=42000C
a=rtpmap:105 H264/90000
a=ftmp:105 profile-level-id=42800a
a=rtpmap:106 H264/90000
a=fmtp:106 profile-level-id=42800c
a=rtpmap:107 H263/90000
a=fmtp:107 QCIF=1;SQCIF=1
```

Message 5: Offer to 3PCR

```
v=0
o=BroadWorks 783 1 IN IP4 10.16.120.22
s=-
c=IN IP4 10.16.120.22
t=0 0
m=audio 25980 RTP/AVP 0 8 18 96 9 2 101
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:96 AMR/8000
a=rtpmap:9 G722/8000
a=rtpmap:2 G726-32/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=ptime:20
a=sendonly
a=label:1
m=video 25978 RTP/AVP 104
b=AS:512
a=rtpmap:104 H264/90000
a=fmtp:104 profile-level-id=42E00D; packetization-mode=0a=sendOnly
a=label:3
m=audio 26042 RTP/AVP 0 8 18 96 9 2 101
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:96 AMR/8000
a=rtpmap:9 G722/8000
a=rtpmap:2 G726-32/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=ptime:20
```

```
a=sendonly
a=label:2
m=video 26040 RTP/AVP 104
b=AS:512
a=rtpmap:104 H264/90000
a=fmtp:104 profile-level-id=42E00D; packetization-mode=0
a=sendOnly
a=label:4
```

Message 6: Answer from 3PCR (CrSDP3/CrSDP4)

```
v=0
o=SIPP 144 0 IN IP4 10.16.134.17
s=Call Recording SDP
t=0 0
m=audio 6008 RTP/AVP 0 18 9
c=IN IP4 10.16.134.17
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=label:1
m=video 6004 RTP/AVP 104 b=AS:512
a=rtpmap:104 H264/90000
a=fmtp:104 profile-level-id=42E00D; packetization-mode=0
a=label:3
m=audio 6010 RTP/AVP 0 18 9
c=IN IP4 10.16.134.17
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=label:2
m=video 6006 RTP/AVP 104
b=AS:512
a=rtpmap:104 H264/90000
a=fmtp:104 profile-level-id=42E00D; packetization-mode=0
a=label:4
```

Message 8: Answer to Media Server (CrSDP3)

```
v=0
o=BroadWorks 785 1 IN IP4 10.16.134.17
s=-
t=0 0
m=audio 6008 RTP/AVP 0 18 9
c=IN IP4 10.16.134.17
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=label:1
m=video 6004 RTP/AVP 104
c=IN IP4 10.16.134.17
b=AS:512
a=rtpmap:104 H264/90000
a=fmtp:104 profile-level-id=42E00D; packetization-mode=0
a=label:2
```

Message 9: Answer to Media Server (CrSDP4)

```
v=0
o=BroadWorks 785 1 IN IP4 10.16.134.17
s=-
t=0 0
m=audio 6010 RTP/AVP 0 18 9
c=IN IP4 10.16.134.17
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=label:3
m=video 6006 RTP/AVP 104
c=IN IP4 10.16.134.17
b=AS:512
a=rtpmap:104 H264/90000
a=fmtp:104 profile-level-id=42E00D; packetization-mode=0
a=label:4
```

Message 10: Application Server to User A

```
 INVITE A (No SDP)
```

Message 11: Offer from User A

```
v=0
o=- 1360011625 1360011626 IN IP4 10.16.134.101
s=Polycom IP Phone
c=IN IP4 10.16.134.101
t=0 0
m=audio 2228 RTP/AVP 0 127
a=rtpmap:0 PCMU/8000
a=rtpmap:127 telephone-event/8000
m=video 2224 RTP/AVP 109 110 34
b=AS:512
a=sendrecv
a=rtpmap:109 H264/90000
a=fmtp:109 profile-level-id=42E00C; packetization-mode=0
a=rtpmap:110 H264/90000
a=fmtp:110 profile-level-id=42800d;max-mbps=40500;max-fs=1344
a=rtpmap:34 H263/90000
a=fmtp:34 QCIF=1;SQCIF=1
```

Message 12: Offer to Media Server

```
v=0
o=- 1360011625 1360011626 IN IP4 10.16.134.101
s=Polycom IP Phone
c=IN IP4 10.16.134.101
t=0 0
m=audio 2228 RTP/AVP 0 127
a=rtpmap:0 PCMU/8000
a=rtpmap:127 telephone-event/8000
m=video 2224 RTP/AVP 109 110
b=AS:512
a=sendrecv
a=rtpmap:109 H264/90000
a=fmtp:109 profile-level-id=42E00C; packetization-mode=0
a=rtpmap:110 H264/90000
a=fmtp:110 profile-level-id=42800d;max-mbps=40500;max-fs=1344
```

Message 13: Answer from Media Server (MSSDP1)

```
v=0
o=BroadWks 14661 0 IN IP4 10.16.120.22
s=Media Server SDP
c=IN IP4 10.16.120.22
t=0 0
m=audio 26038 RTP/AVP 0 8 18 96 9 2 101
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:96 AMR/8000
a=rtpmap:9 G722/8000
a=rtpmap:2 G726-32/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=ptime:20
m=video 26036 RTP/AVP 109
a=rtpmap:109 H264/90000
a=fmtp:109 profile-level-id=42E00C; packetization-mode=0
```

Message 15: Application Server to Media Server

```
INVITE to MS (NO SDP)
```

Message 16: Answer from Media Server (MSSDP2)

```
v=0
o=BroadWks 14660 0 IN IP4 10.16.120.22
s=Media Server SDP
c=IN IP4 10.16.120.22
t=0 0
m=audio 26046 RTP/AVP 0 8 18 96 9 2 101
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:96 AMR/8000
a=rtpmap:9 G722/8000
a=rtpmap:2 G726-32/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=ptime:20
m=video 26044 RTP/AVP 104 105 106 107
a=rtpmap:104 H264/90000
a=fmtp:104 profile-level-id=42E00C; packetization-mode=0
a=rtpmap:105 H264/90000
a=ftmp:105 profile-level-id=42800a
a=rtpmap:106 H264/90000
a=fmtp:106 profile-level-id=42800c
a=rtpmap:107 H263/90000
a=fmtp:107 QCIF=1;SQCIF=1
```

Message 17: INVITE to User B (SDPA with MSSDP2 address/port)

```
v=0
o=BroadWorks 787 1 IN IP4 10.16.120.22
s=-
c=IN IP4 10.16.120.22
t=0 0
m=audio 26046 RTP/AVP 0 8 18 96 9 2 101
a=rtpmap:0 PCMU/8000
```

```
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:96 AMR/8000
a=rtpmap:9 G722/8000
a=rtpmap:2 G726-32/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=ptime:20
m=video 26044 RTP/AVP 104
b=AS:512
a=sendrecv
a=rtpmap:104 H264/90000
a=fmtp:104 profile-level-id=42E00D; packetization-mode=0
```

Message 18: Answer from User B (SDPB)

```
v=0
o=- 1360011671 1360011672 IN IP4 10.16.134.100
s=Polycom IP Phone
c=IN IP4 10.16.134.100
t=0 0
m=audio 2230 RTP/AVP 0 101
a=rtpmap:0 PCMU/8000
a=rtpmap:101 telephone-event/8000
m=video 2226 RTP/AVP 104
b=AS:512
a=sendrecv
a=rtpmap: 104 H264/90000
a=fmtp:104 profile-level-id=42E00D; packetization-mode=0
```

Message 20: Answer to Media Server (SDPB)

```
v=0
o=BroadWorks 782 2 IN IP4 10.16.134.100
s=-
c=IN IP4 10.16.134.100
t=0 0
m=audio 2230 RTP/AVP 0 101
a=rtpmap:0 PCMU/8000
a=rtpmap:101 telephone-event/8000
m=video 2226 RTP/AVP 104
b=AS:512
a=sendrecv
a=rtpmap:104 H264/90000
a=fmtp:104 profile-level-id=42E00D; packetization-mode=0
```

Message 21: Answer to User A with SDPB and MSSDP1

```
v=0
o=BroadWorks 786 1 IN IP4 10.16.120.22
s=-
c=IN IP4 10.16.120.22
t=0 0
m=audio 26038 RTP/AVP 0 8 18 96 9 2 101
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:96 AMR/8000
a=rtpmap:9 G722/8000
a=rtpmap:2 G726-32/8000
a=rtpmap:101 telephone-event/8000
```

```
a=fmtp:101 0-15
a=ptime:20
m=video 26036 RTP/AVP 109
b=AS:512
a=sendrecv
a=rtpmap:109 H264/90000
a=fmtp:109 profile-level-id=42E00D; packetization-mode=0
```

10.3.4 Basic Video Call Recording in Single Mode

Following is the call flow for basic single mode video call recording.

The call is established between Users A and B with the SDP negotiated for audio and video from end to end with the resulting video codec being H.264. The Cisco BroadWorks Telephony Application Server then establishes slow start sessions on the Media Server, which are then sent to the 3PCR platform, but with the video SDP information from Users A and B. The audio codec information sent to the 3PCR is ordered to contain the negotiated codec as the first element on the list. The video codec is also updated to have H.264 as the only one on the list as received from the Media Server.

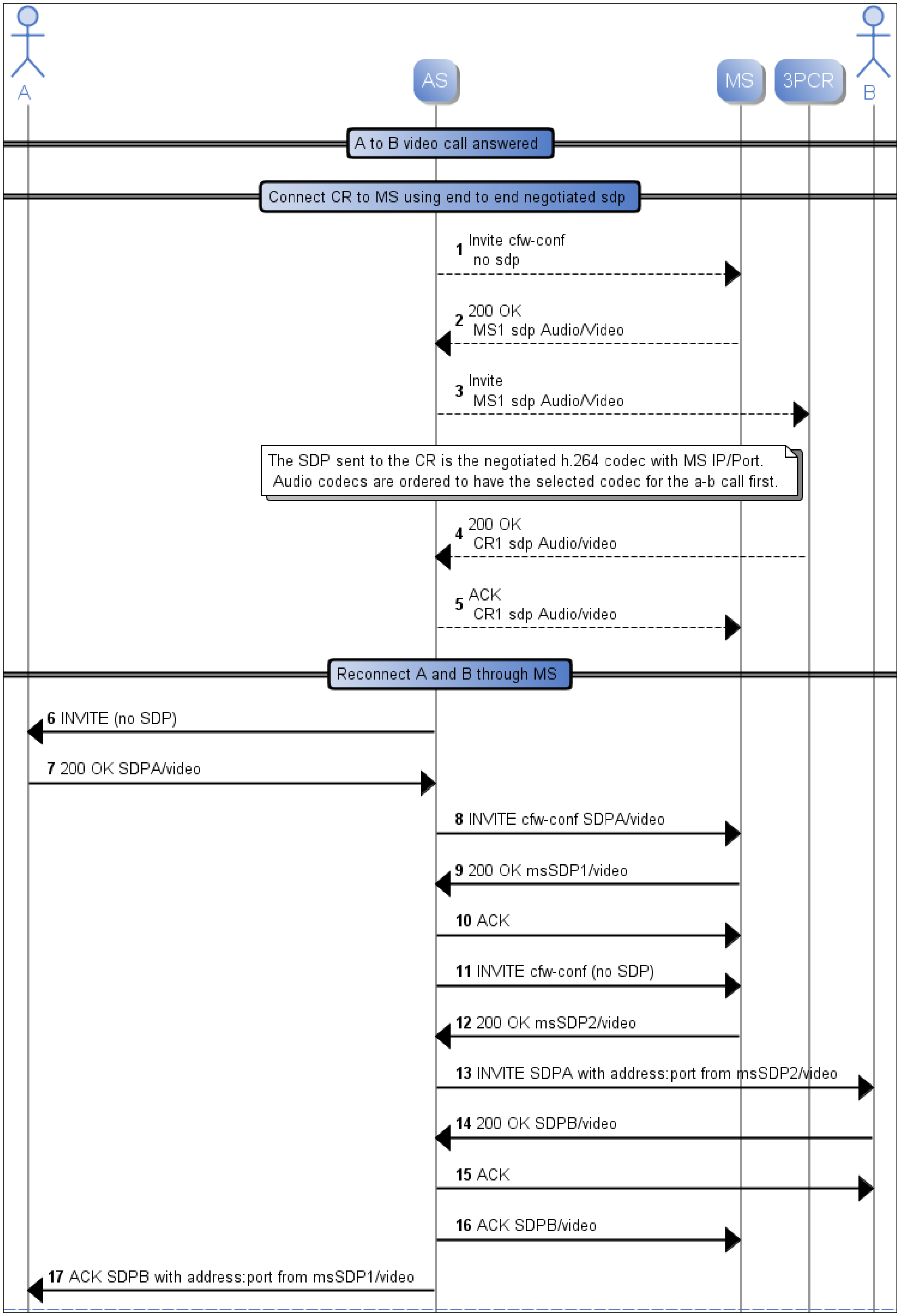

Figure 36 Video Recording in Single Mode (a)

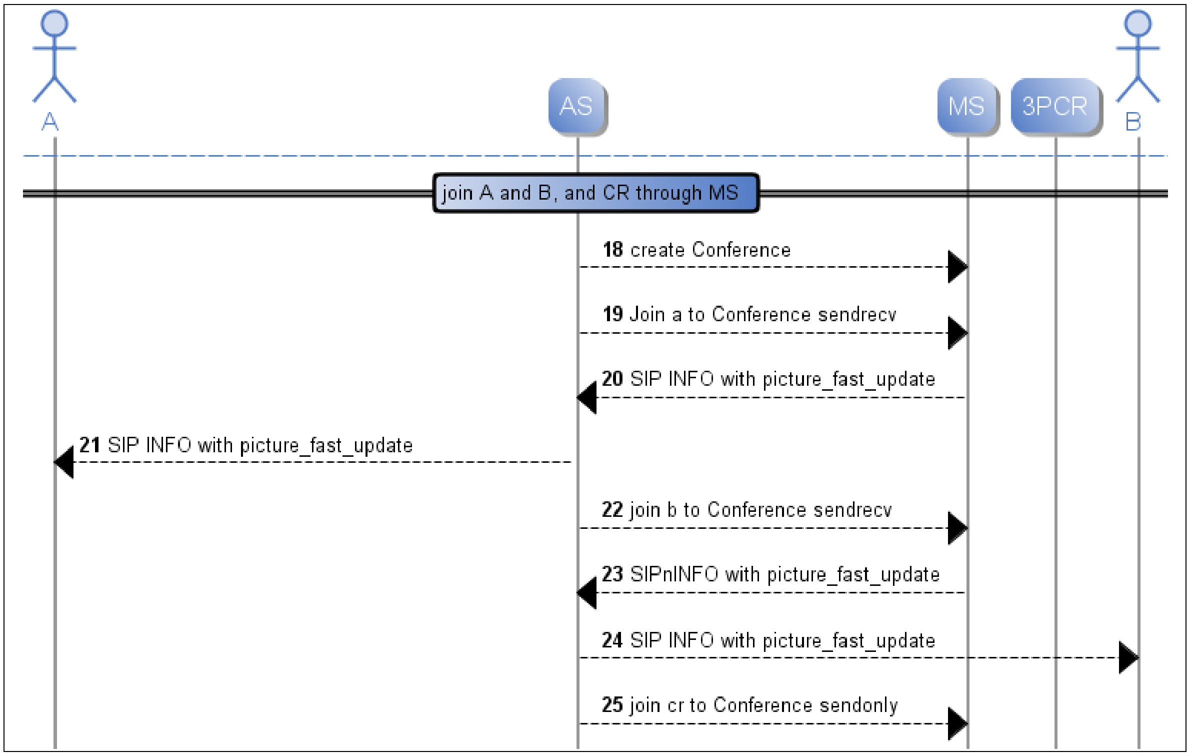

Figure 37 Video Recording in Single Mode (b)

10.3.5 Video Add Mid-Session in Dual Mode

When an audio recording is already in progress, video can be negotiated between end users, and if call recording is already active on that call, then video is also added to the same call recording session.

At this point in the call, Users A and B are already connected through the Media Server for call recording. Then User A adds video to the call and re-invites User B with video added to the SDP. The offer is accepted by User B and video is streamed directly between Users A and B while audio is being streamed through the Media Server. The Call Recording service puts the 3PCR on hold and reconnects to add video media to the same call recording session so that both audio and video are streamed through the Media Server.

The following figures show the call flow for video added mid-session in dual mode.

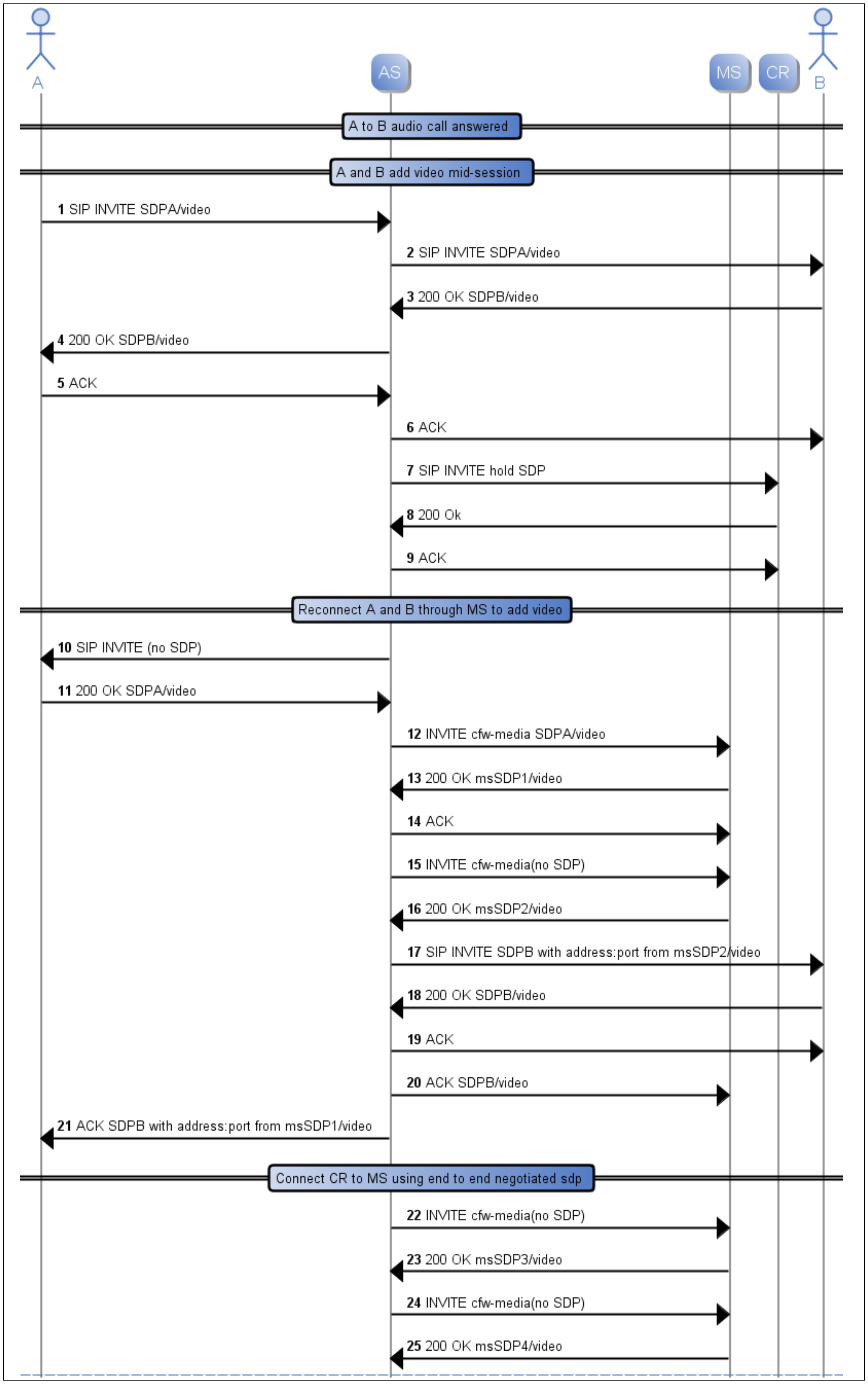

Figure 38 Video Add Mid-Session in Dual Mode (a)

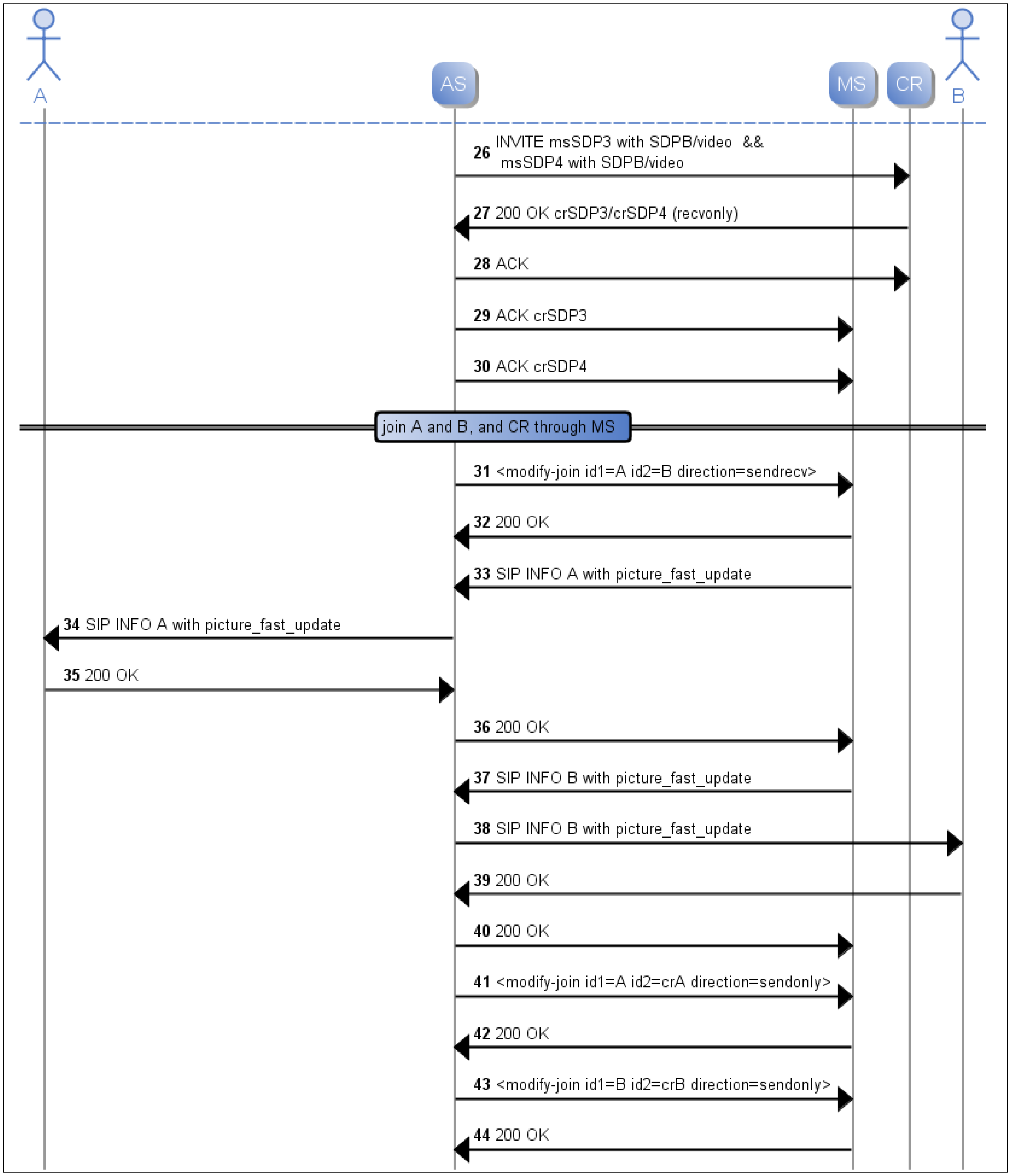

Figure 39 Video Add Mid-Session in Dual Mode (b)

10.3.6 Video Add Mid-Session in Single Mode

This is the call flow for adding video mid-session in single mode. Note that in this situation, the existing sessions with the 3PCR are not released. They are re-invited to add video to the recording sessions.

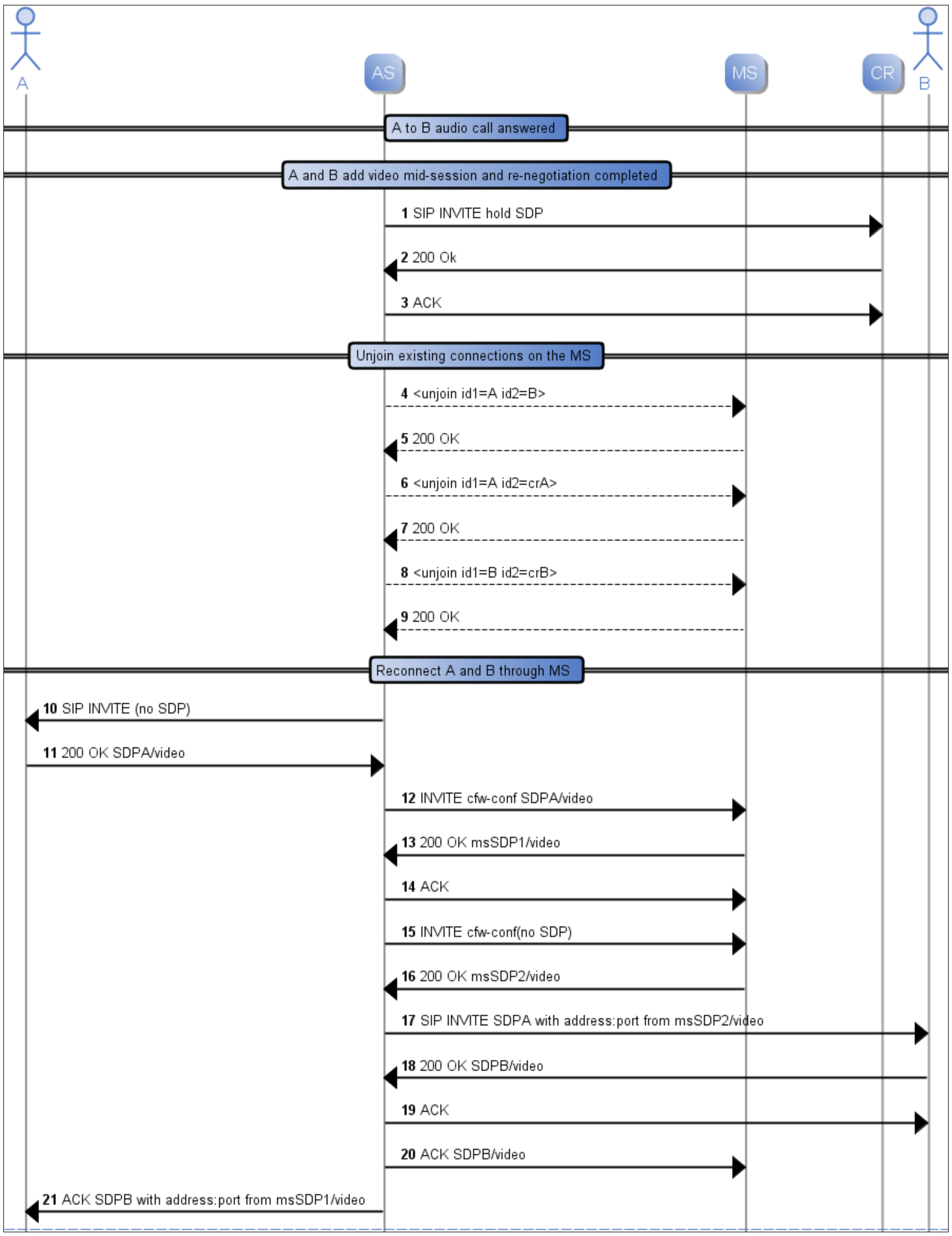

Figure 40 Video Add Mid-Session in Single Mode (a)

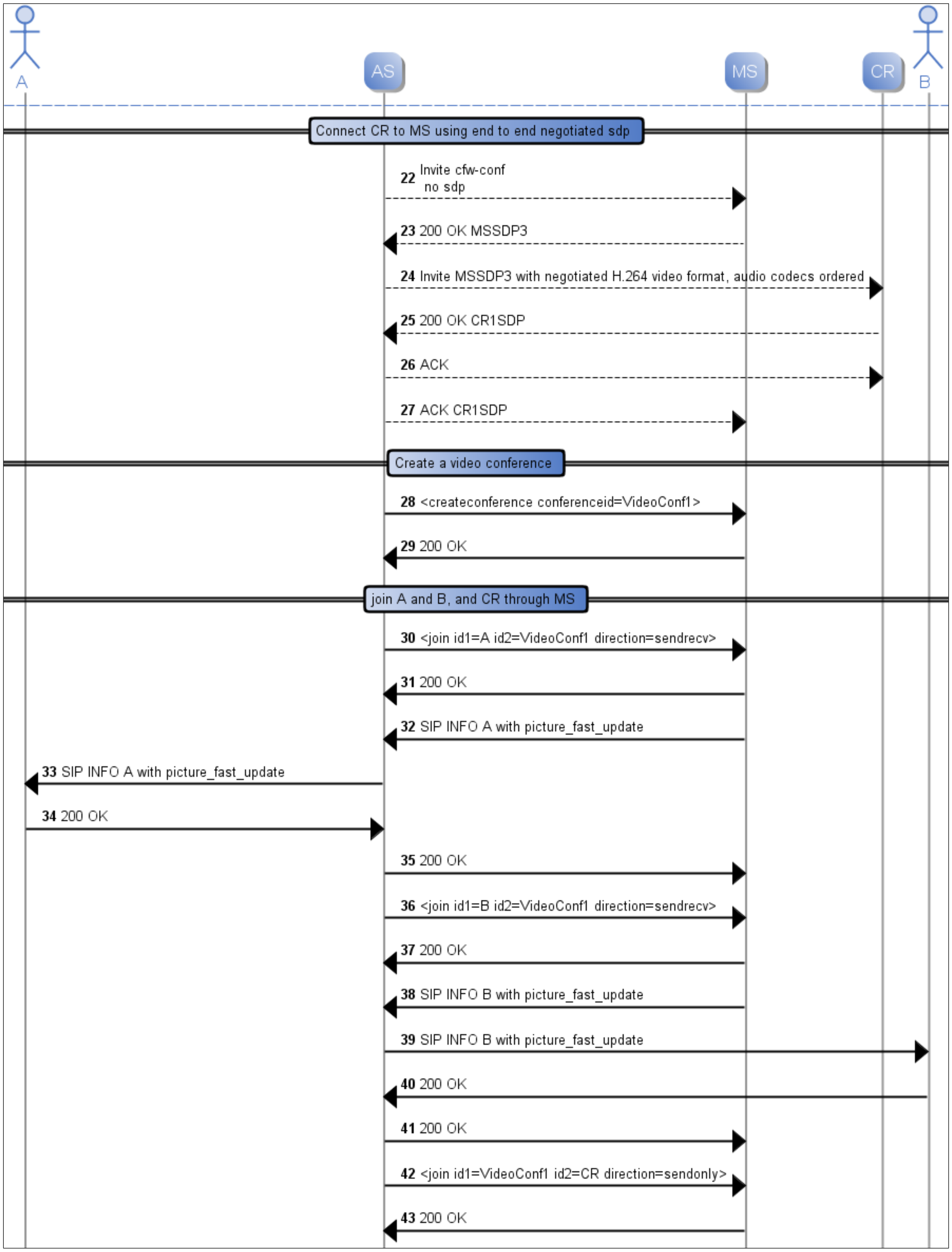

Figure 41 Video Add Mid-Session in Single Mode (b)

10.3.7 Video Conferencing

Following is the call flow for a three-way conference call in dual mode that is device-initiated with Call Recording service for User A.

User A is on a call with User B and call recording is active. User A then puts this call on hold and calls User C. User C answers and call recording is also active for this call. Now User A creates a conference by invoking conferencing on the Cisco BroadWorks Telephony Application Server. The result is a conference created on the Media Server to which the users are connected.

There are two parts to this call. One is to set up the conference between the users and the Media Server. The second is where call recording connections are set up between the user being recorded and the Call Recording platform through the Media Server using the conference bridge from the first part of the call. The first part of the call follows the generic conference call flow on the Application Server. In the second part of the call, User A is connected to the conference.

There are two call recordings created, one for the User A-B call and another for the User A-C call.

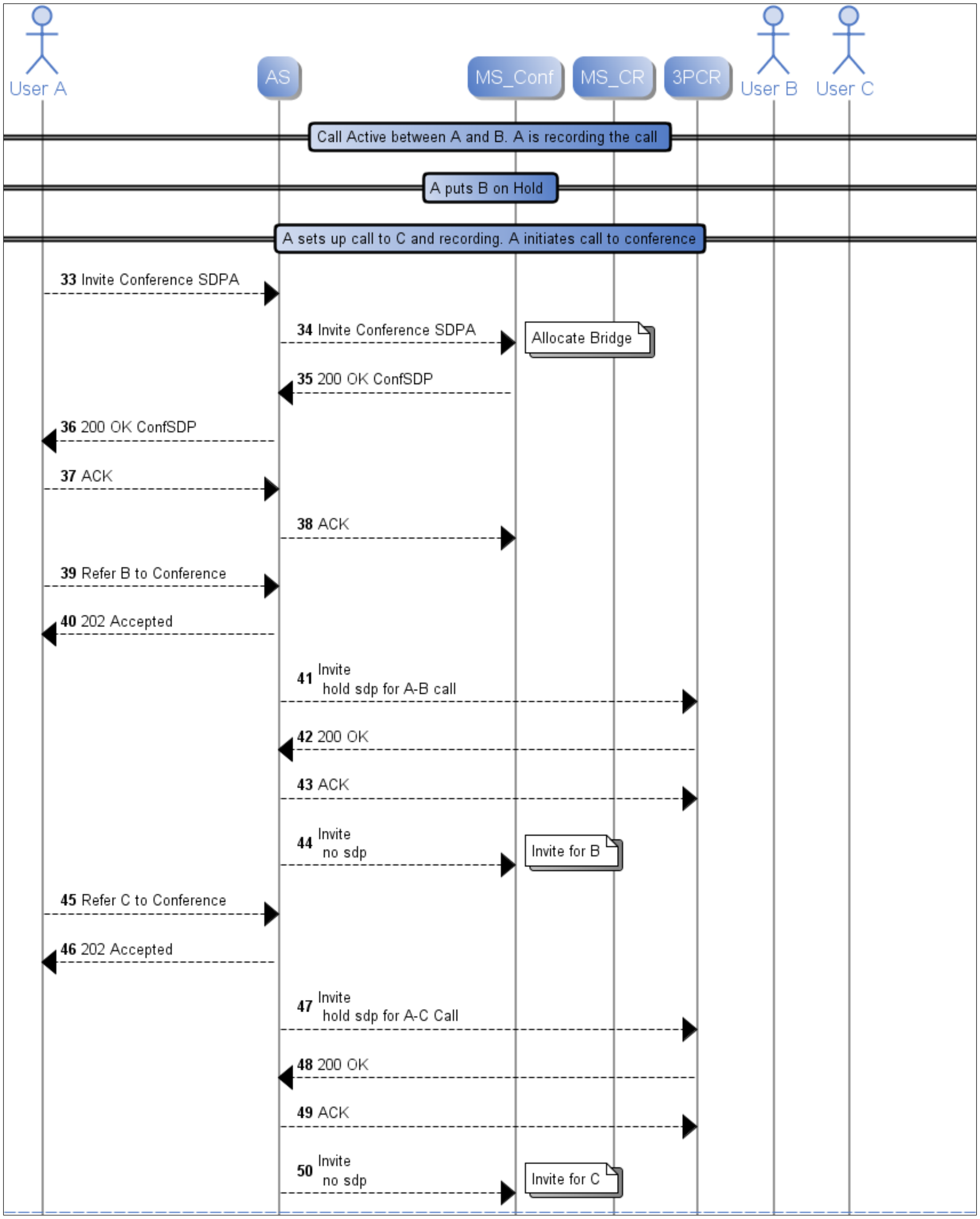

Figure 42 Three-Way Conference Call in Dual Mode (a)

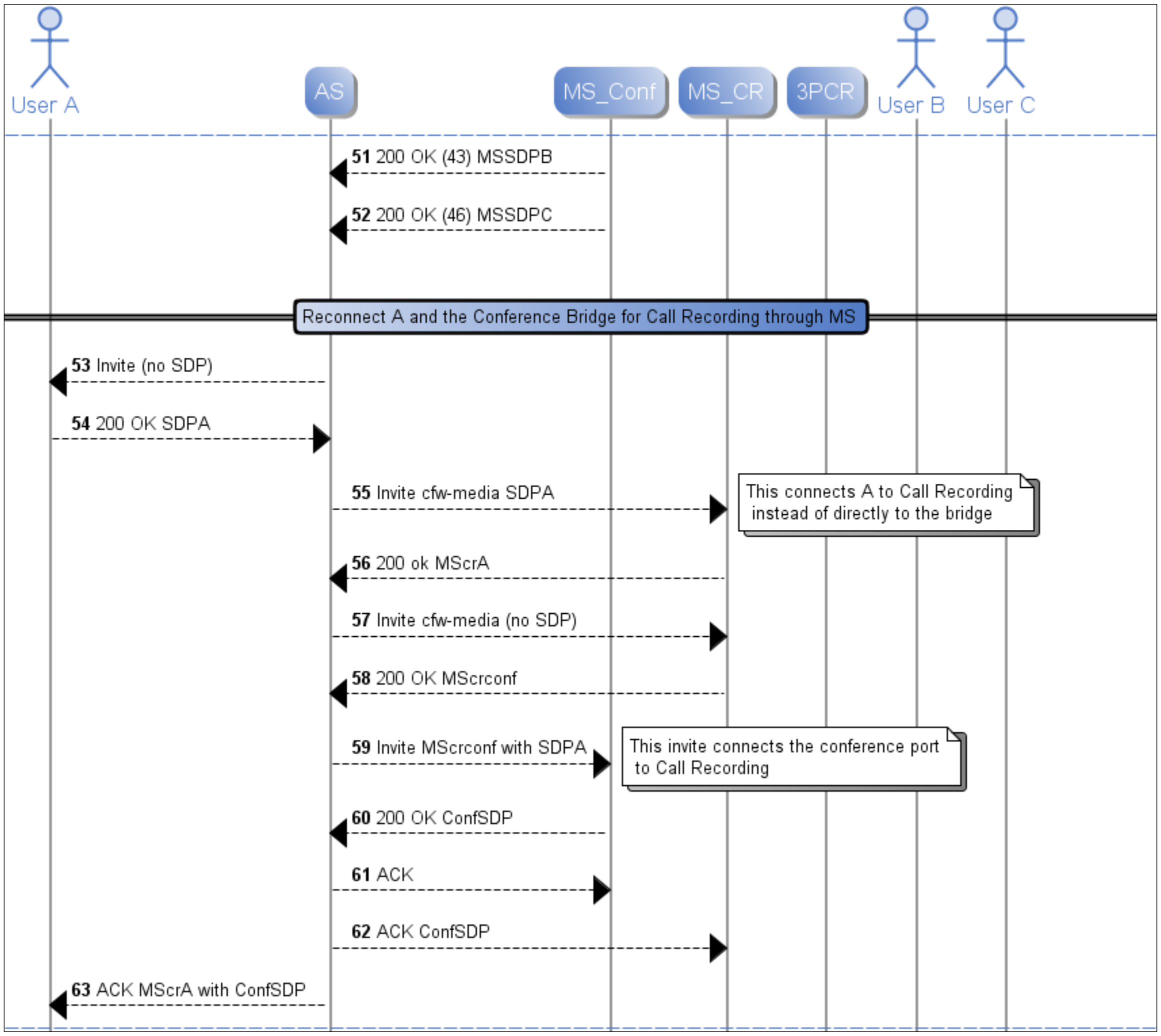

Figure 43 Three-Way Conference Call in Dual Mode (b)

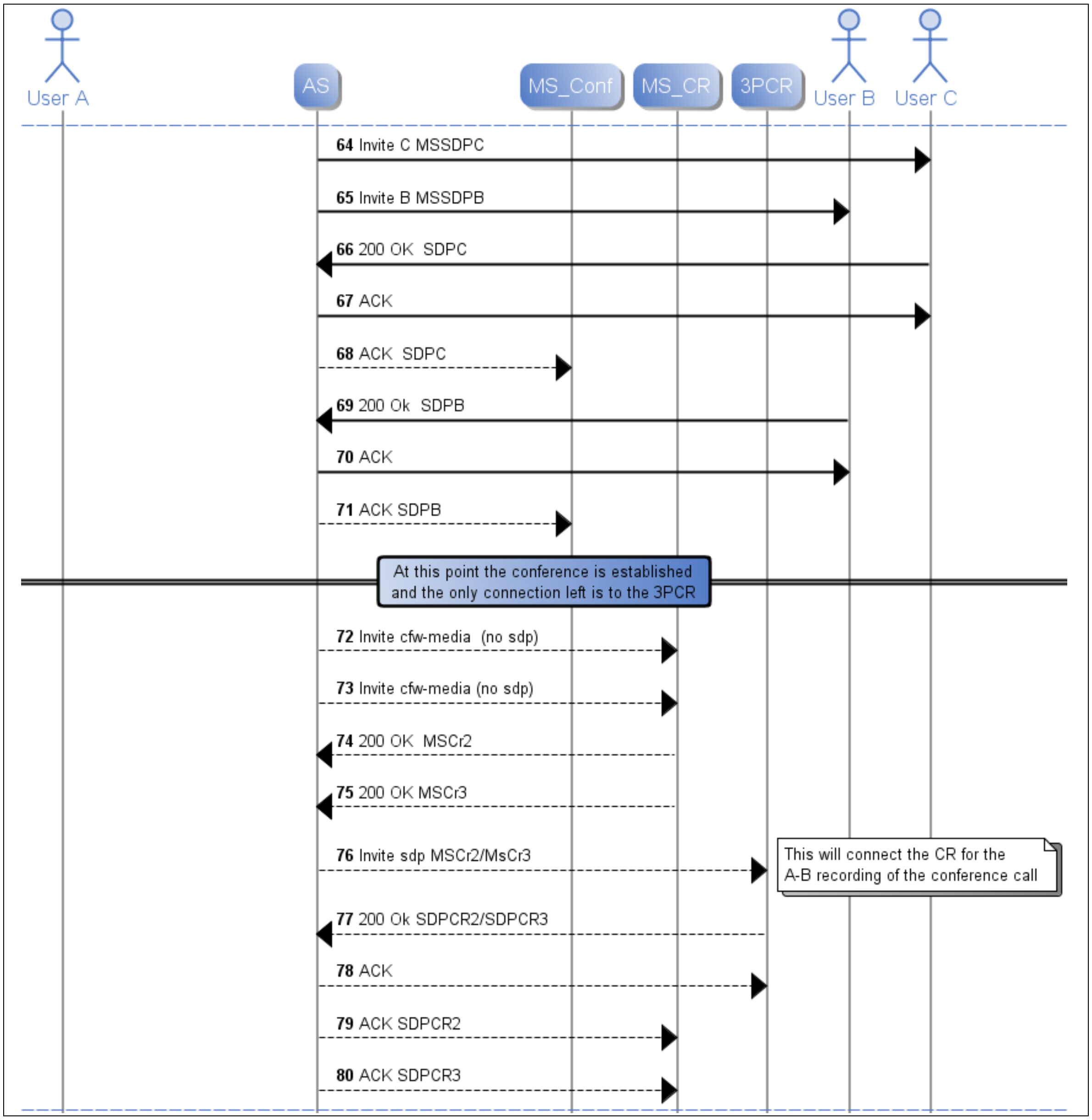

Figure 44 Three-Way Conference Call in Dual Mode (c)

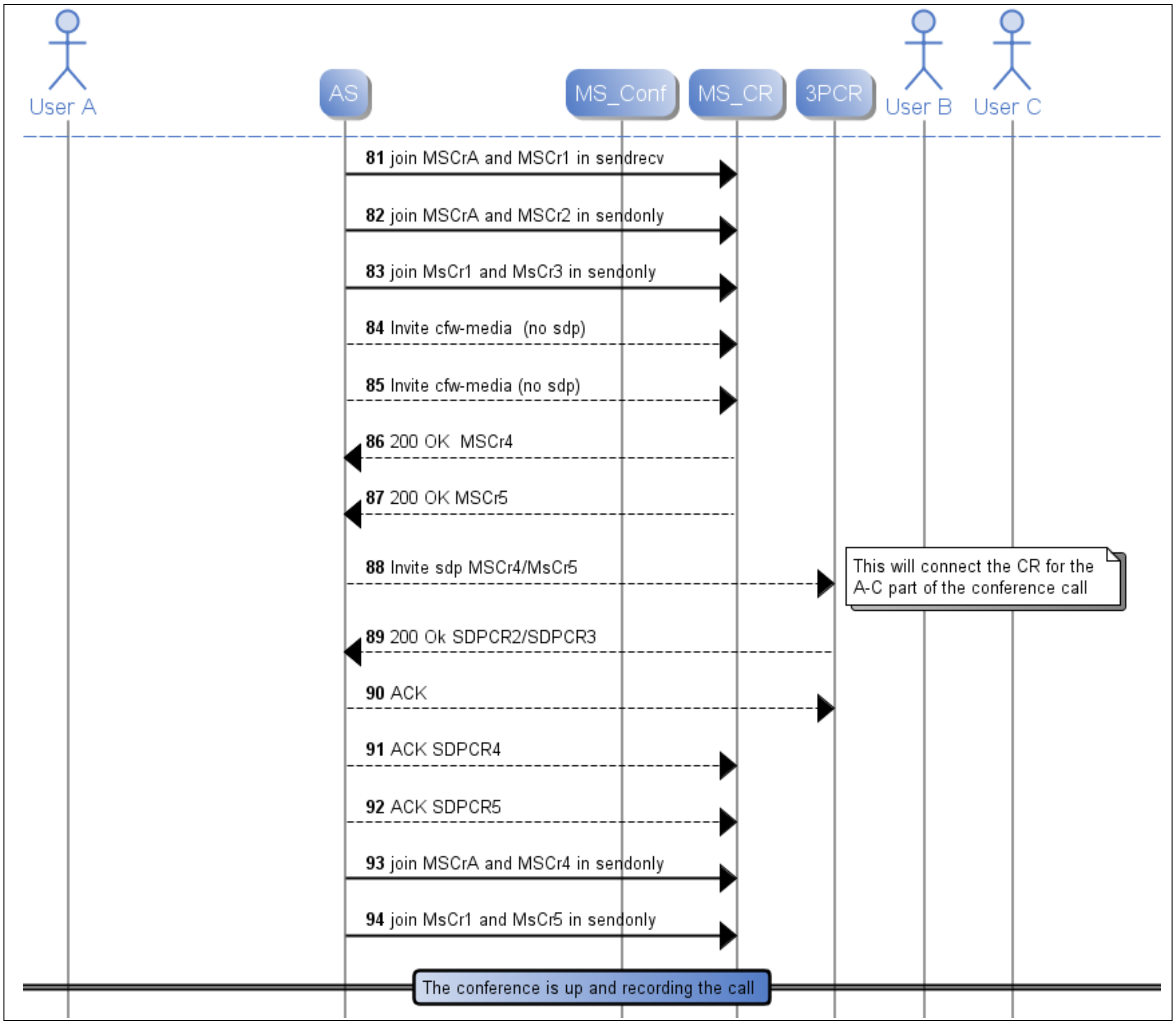

Figure 45 Three-Way Conference Call in Dual Mode (d)

10.3.8 Blind Transfer

In a blind transfer, a user is transferred to another party, and the user performing the transfer does this without talking to the party who is receiving the call. In the following call flow, User B is transferring the call with User A to User C. User A has call recording active and has two call recordings associated with it. One is *before* the transfer for the call between User A and User B and one is *after* the transfer between User A and User C.

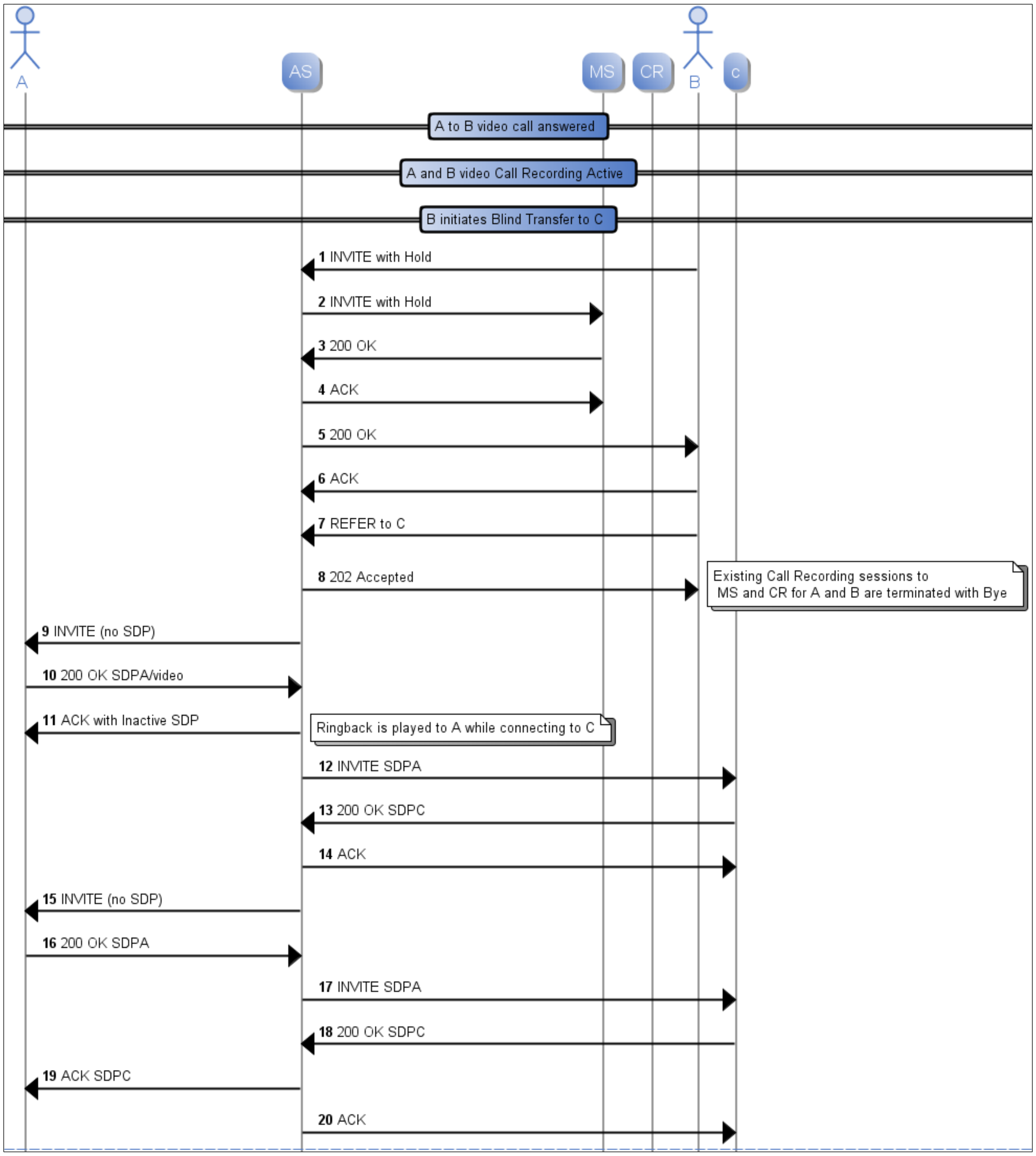

Figure 46 Blind Transfer (a)

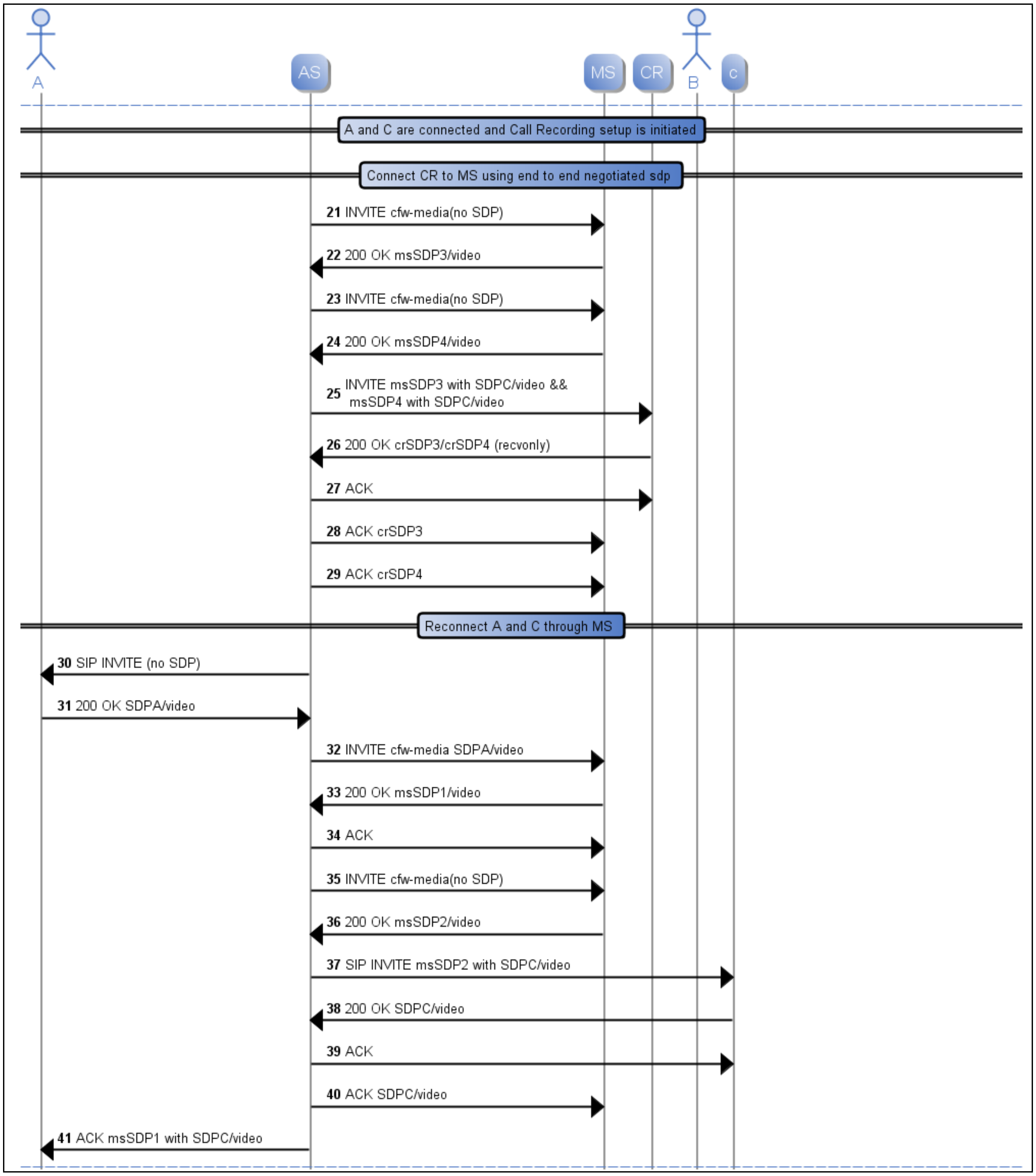

Figure 47 Blind Transfer (b)

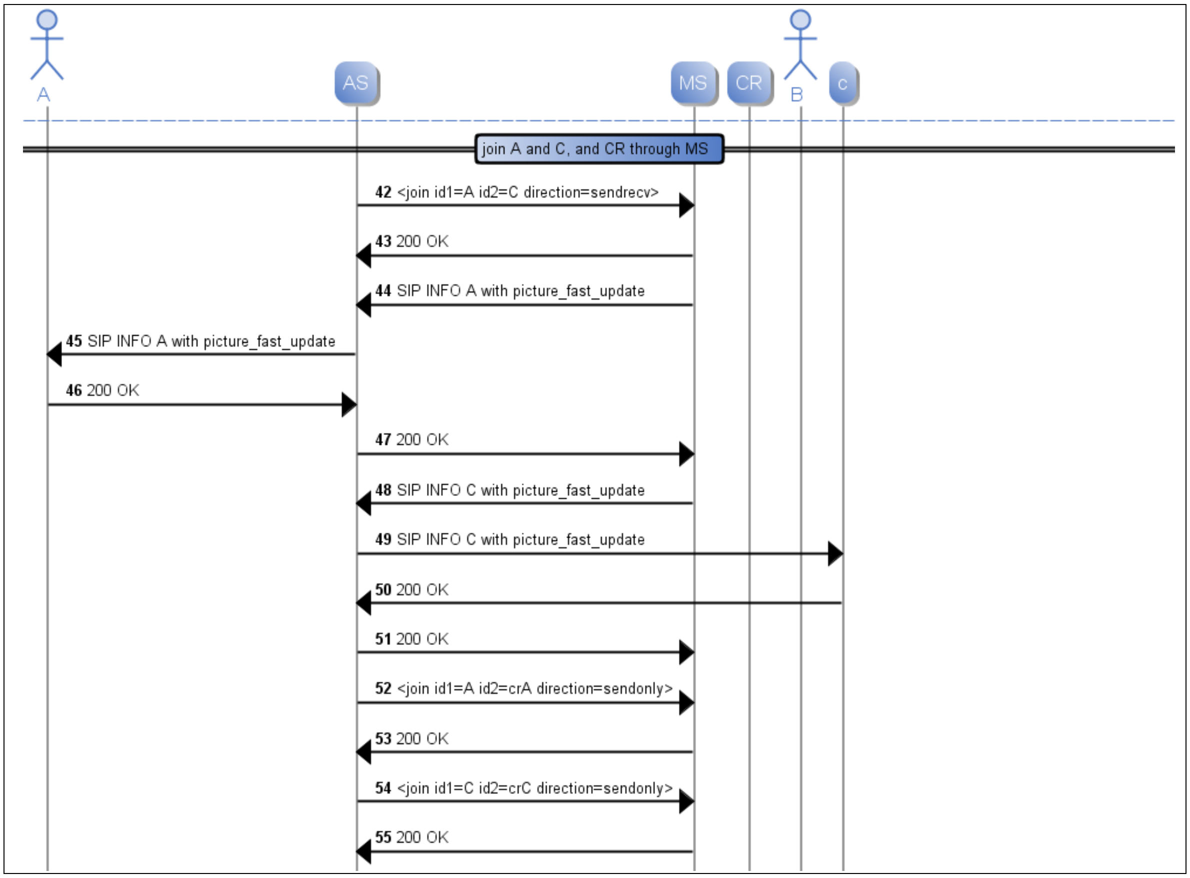

Figure 48 Blind Transfer (c)

10.3.9 Consultative Transfer

In a consultative transfer, the user transferring the call talks to the target of the transfer prior to completing the transfer. Following is a call flow for a consultative transfer in which all users have call recording active before the transfer. For clarity, only the transactions for User A are shown in the call flow. User B is on calls with User A and User C. User B then "consult transfers" User A to User C. Once the transfer is complete, the 3PCR, User A, and User C are reconnected to record audio and video.

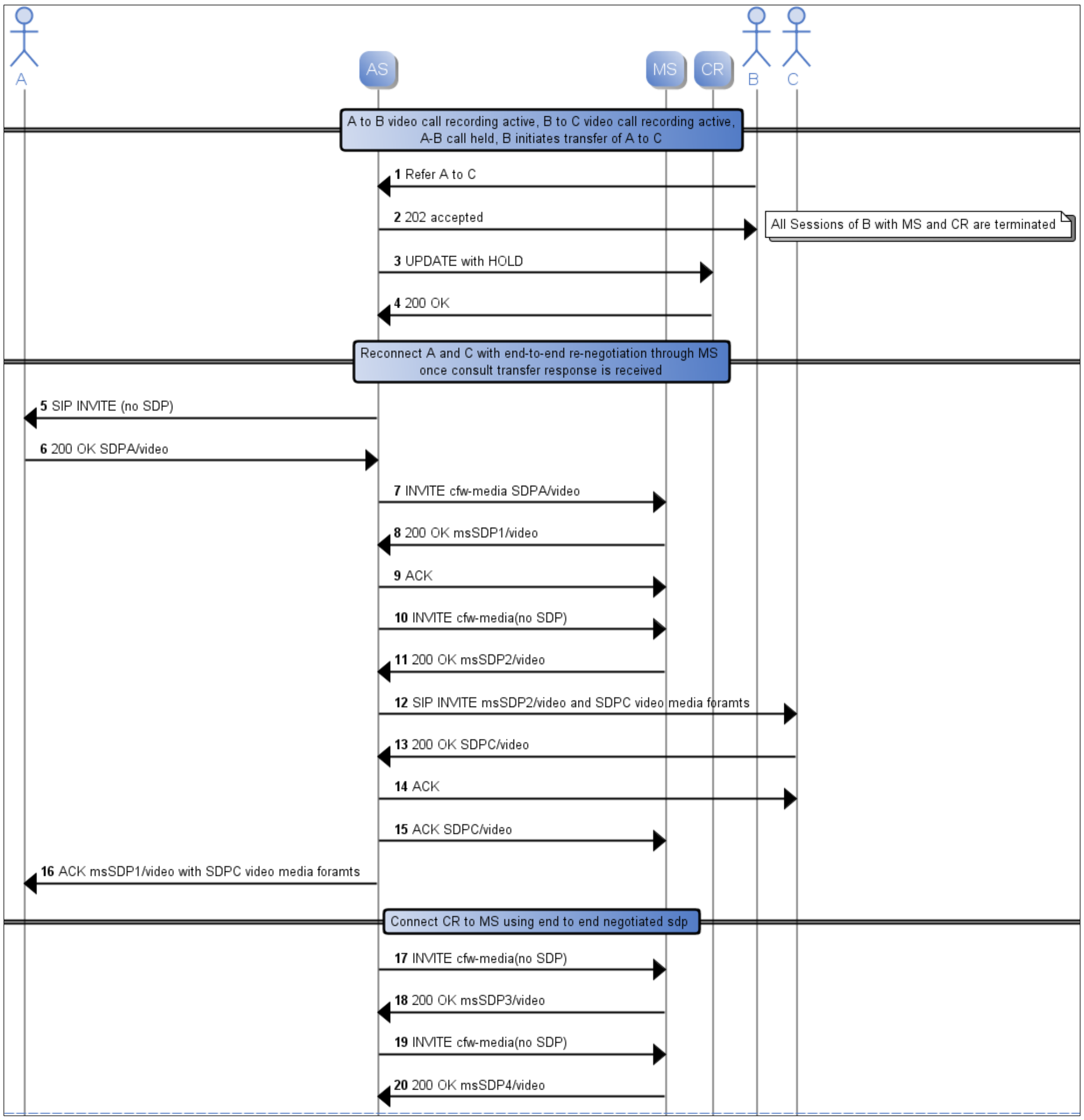

Figure 49 Consultative Transfer (a)

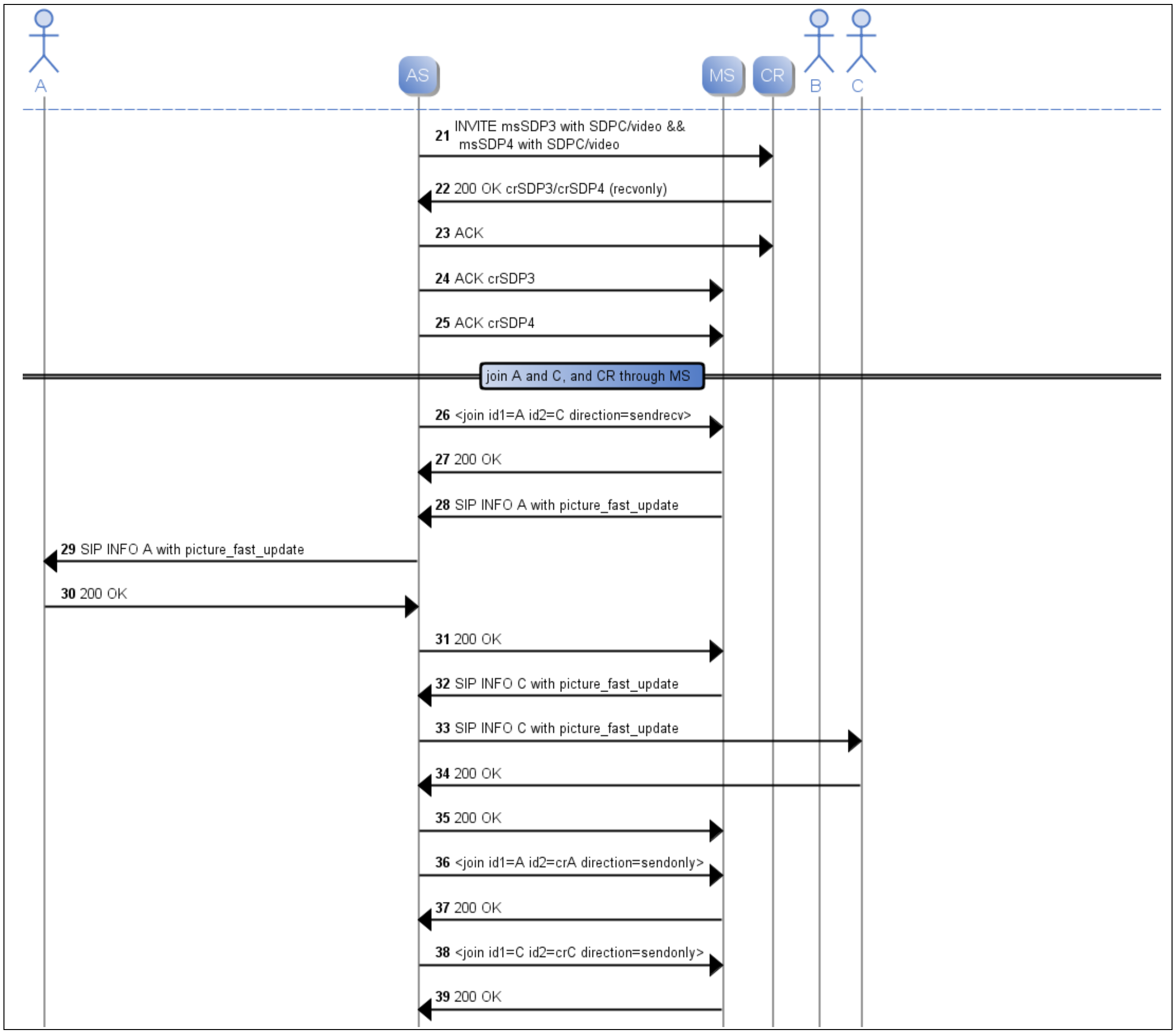

Figure 50 Consultative Transfer (b)

10.4 Voice Mail Recording Call Flows

10.4.1 Voice Mail Deposit Due to Unanswered Call

In the following call flow, User A calls User B. User B does not answer and the call is routed to User B's voice mail.

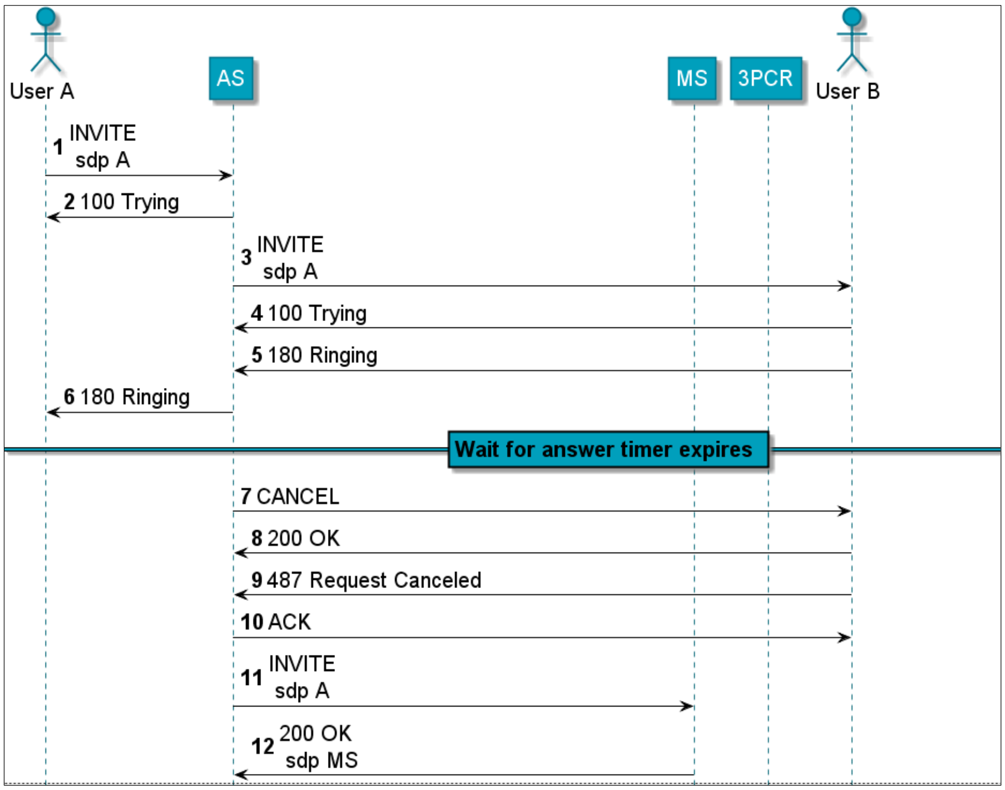

Figure 51 Unanswered Call (a)

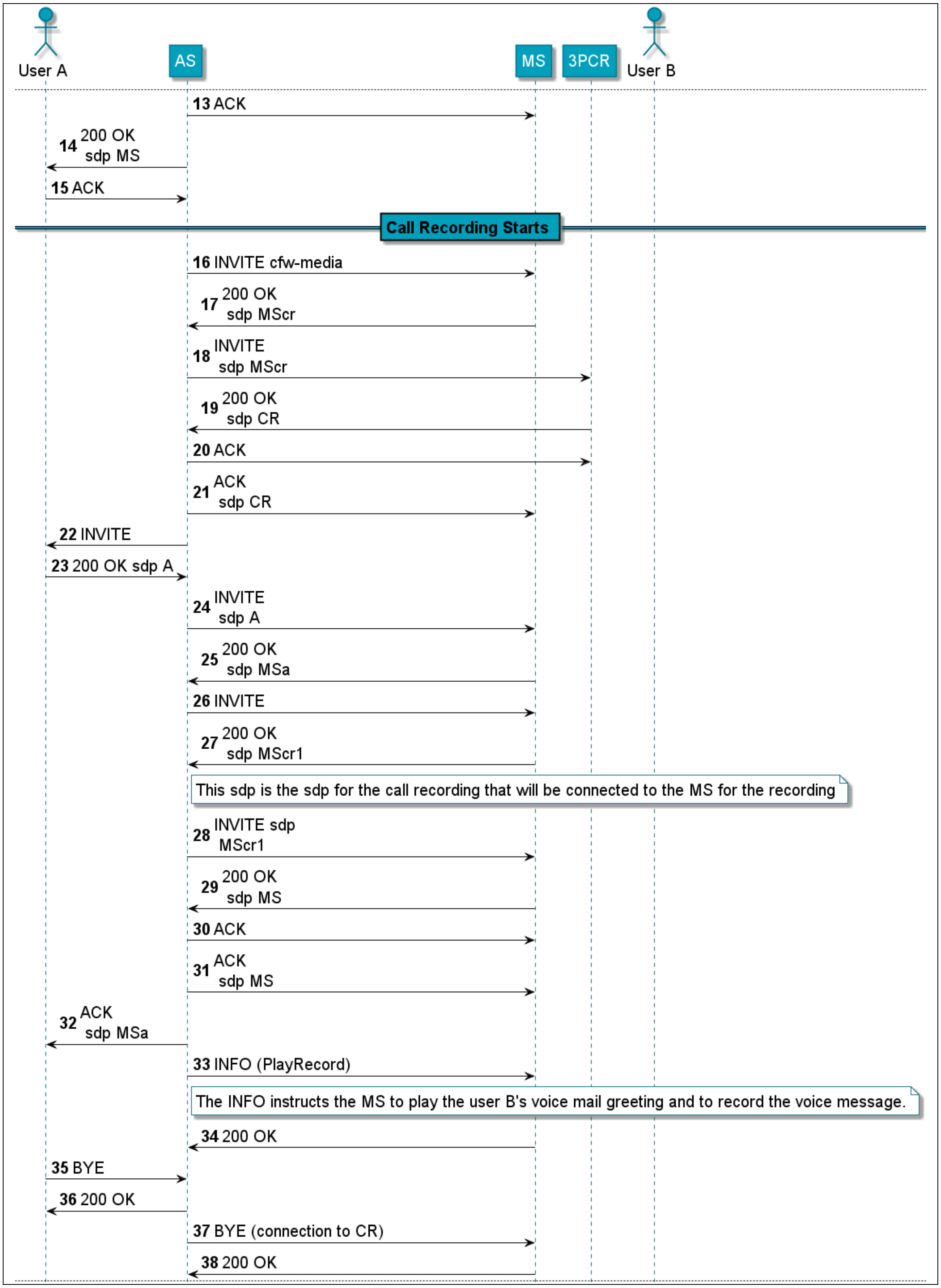

Figure 52 Unanswered Call (b)

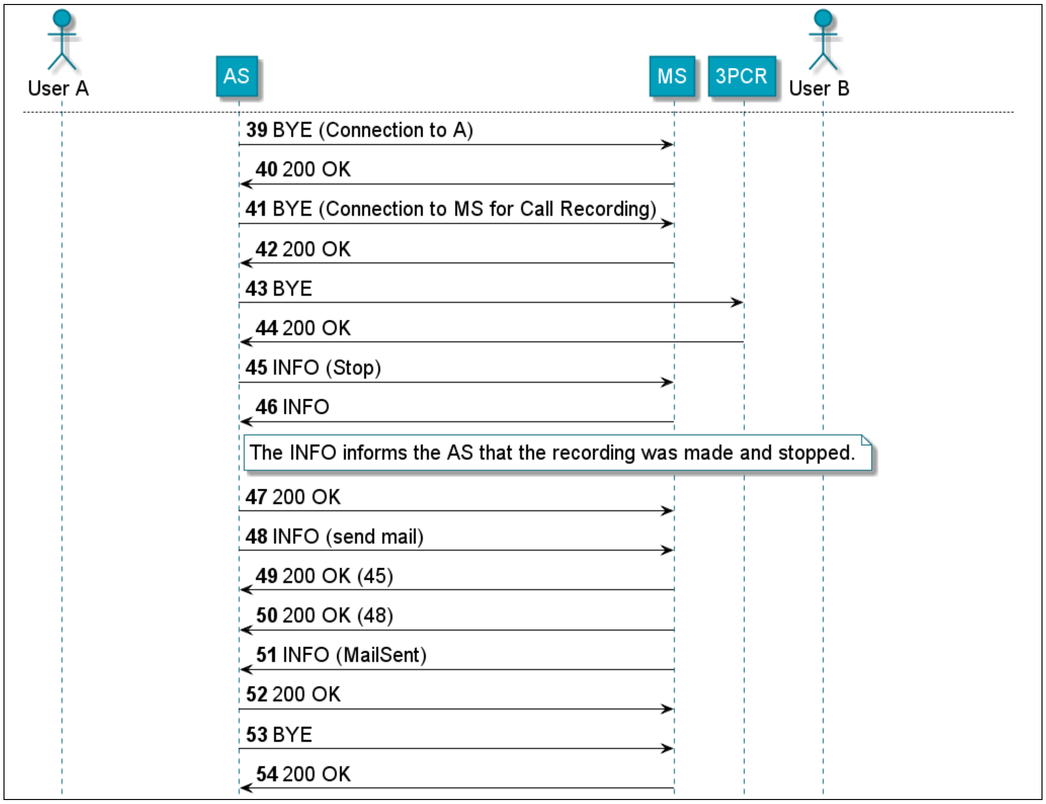

Figure 53 Unanswered Call (c)

10.4.2 Voice Mail Call Recording Abandoned for Voice Portal Access

In the following call flow, the caller dials a number and when the call is routed to voice mail, the caller presses “*” to access the voice portal. Accessing the voice portal causes the Call Recording service to release the connection to the 3PCR platform.

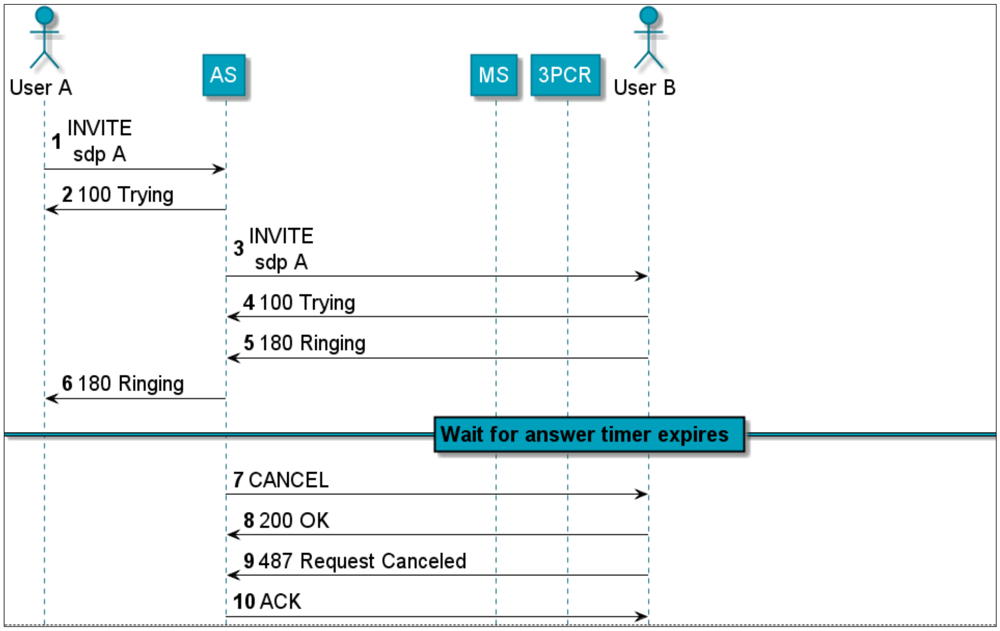

Figure 54 Voice Portal Access (a)

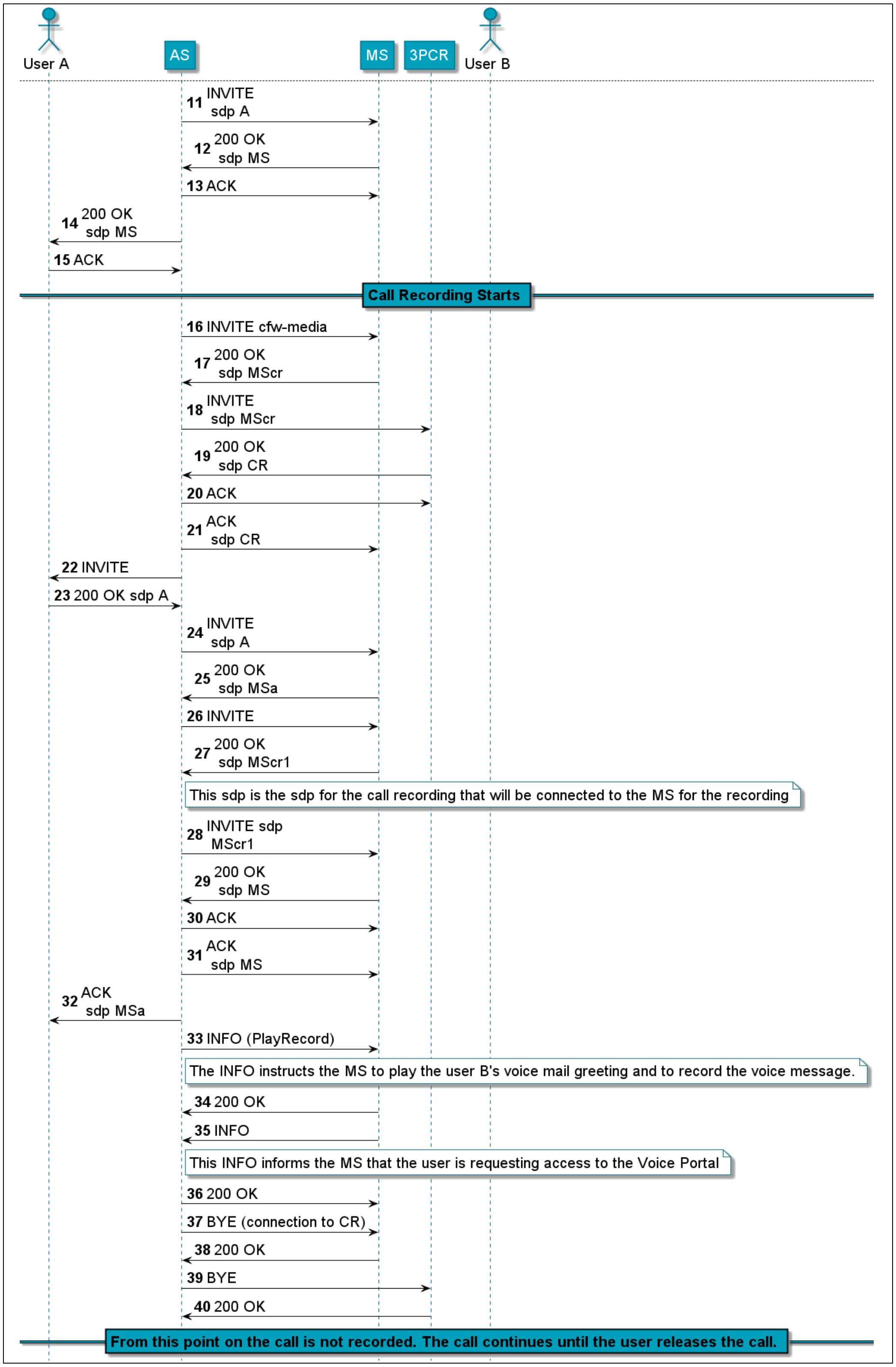

Figure 55 Voice Portal Access (b)

10.4.3 User Deletes First Voice Message and Leaves Another

In the following example, the user deletes the first voice message and records a second one.

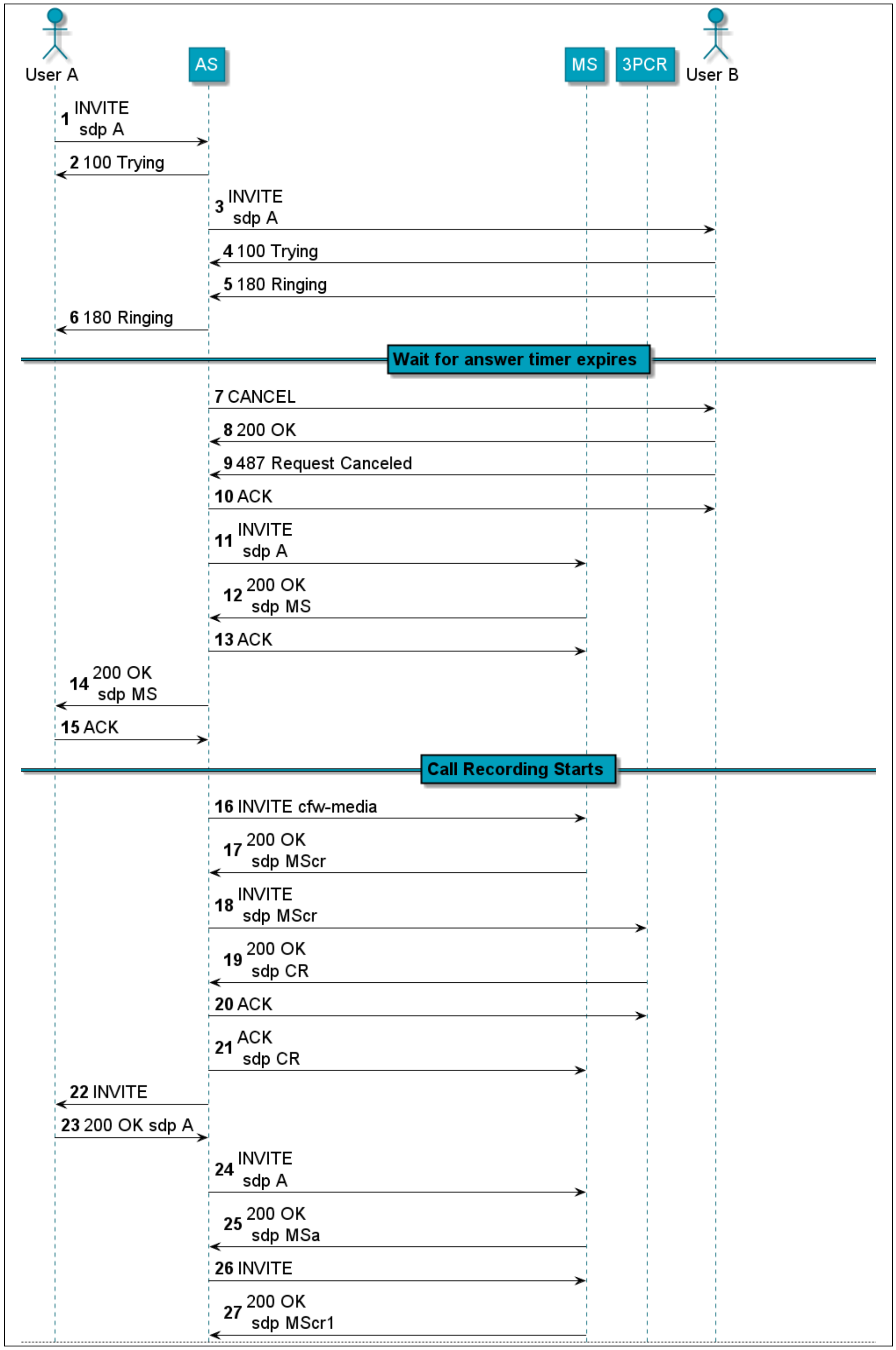

Figure 56 Recording Multiple Messages (a)

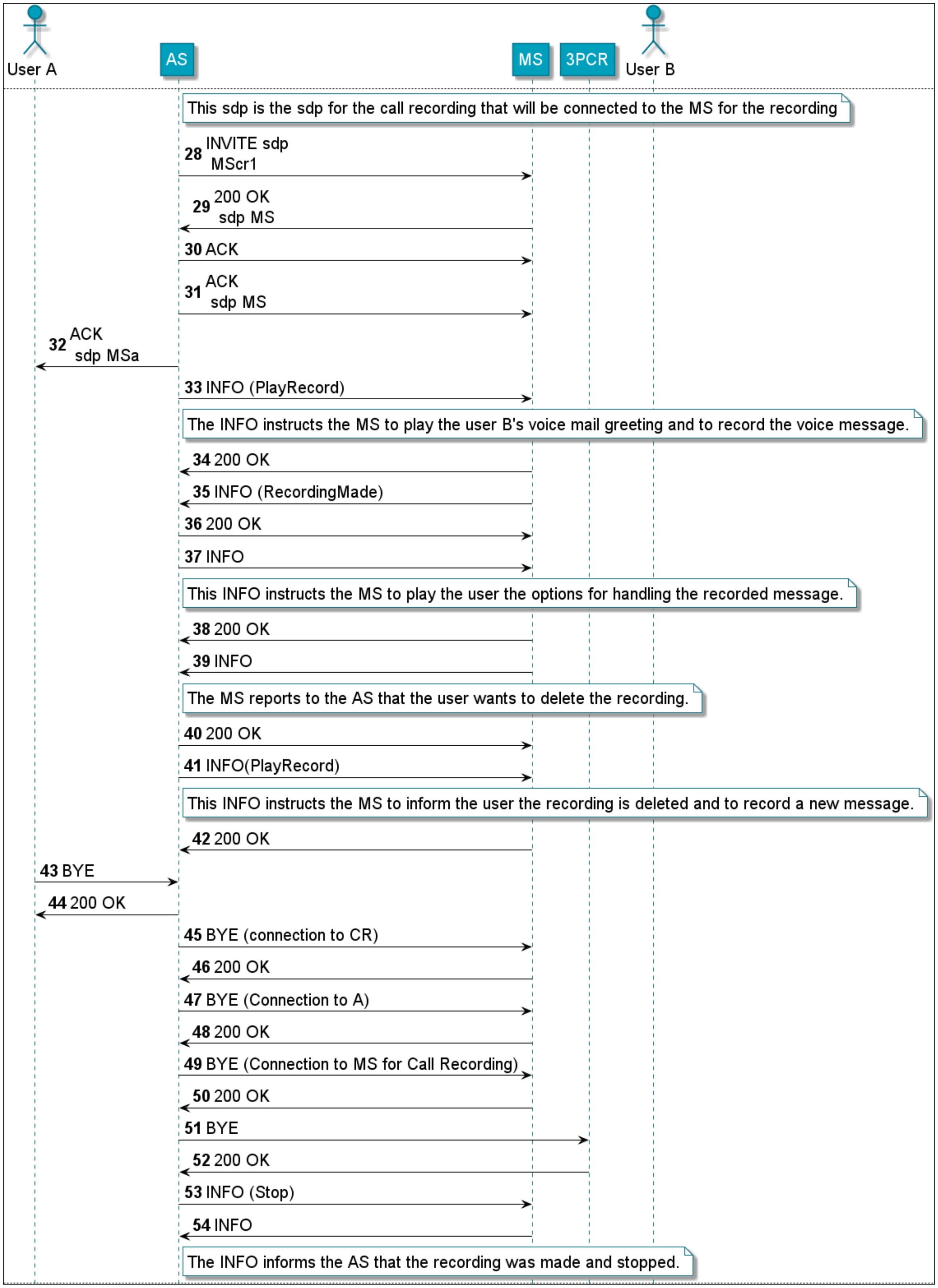

Figure 57 Recording Multiple Messages (b)

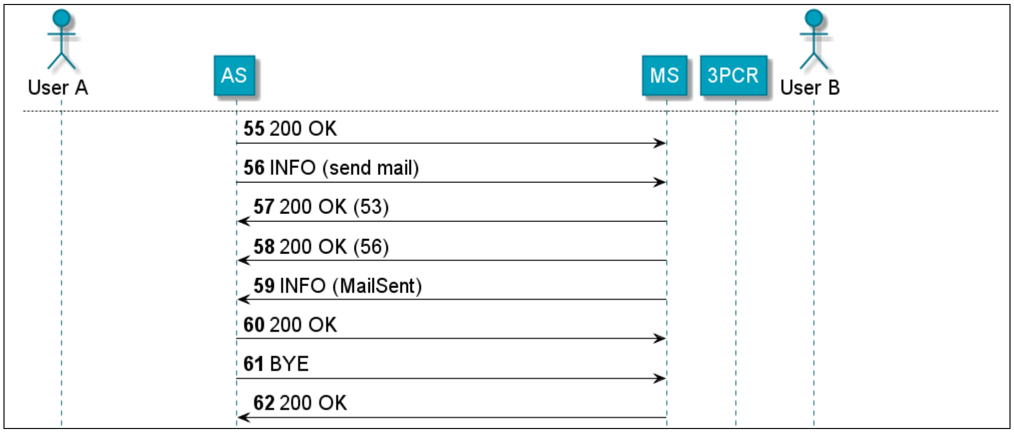

Figure 58 Recording Multiple Messages (c)

10.4.4 Voice Mail Deposit – All Calls Route to Voice Mail

The following example is similar to the first call flow shown in the three figures in section *10.4.1 Voice Mail Deposit Due to Unanswered Call*. The difference is that the user has provisioned their voice messaging to route all calls to voice mail.

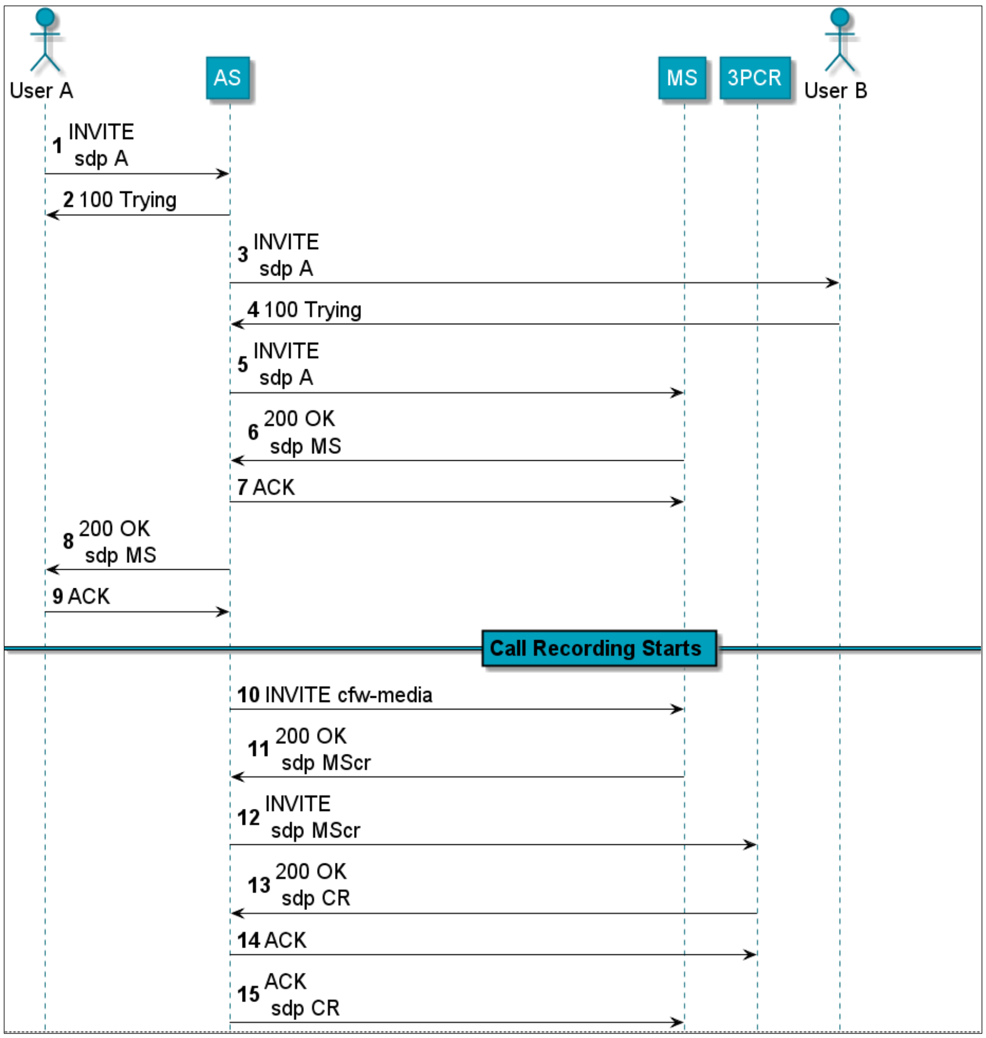

Figure 59 Routing All Calls to Voice Mail (a)

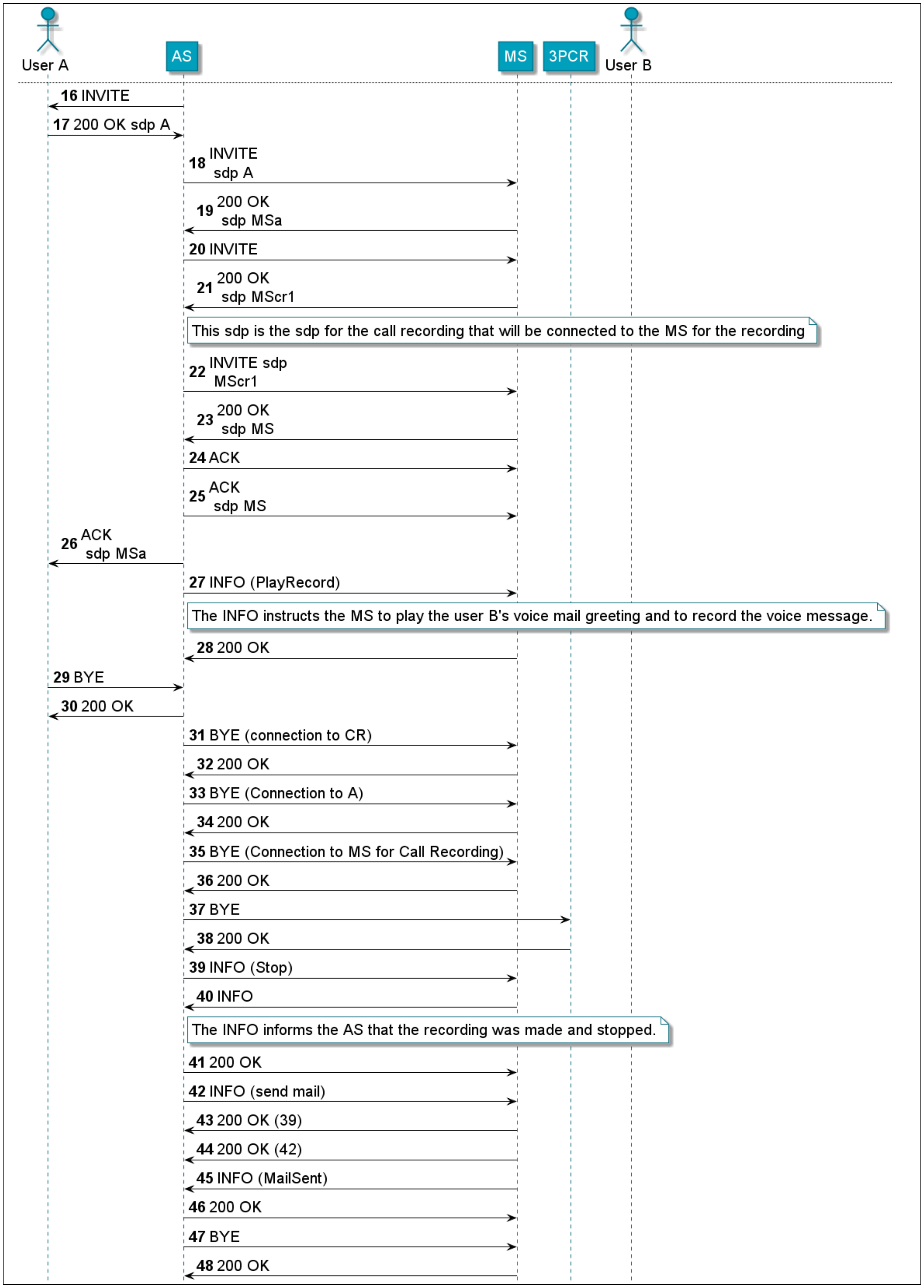

Figure 60 Routing All Calls to Voice Mail (b)

10.5 Terminator Recording Call

This call flow shows the messages when the terminating party has Call Recording enabled. The terminator is using a dual stream to the Call Recording Server. Details of the messages to/from the Call Recording platform appear after the call flow diagram. Note that in this call flow, the two stream SDP descriptions, MS3 and MS4, are combined in the invitation to the 3PCR platform and the resulting response is separated out into CR3 and CR4 going back to the Media Server. The SDP label attribute is used to coordinate these manipulations.

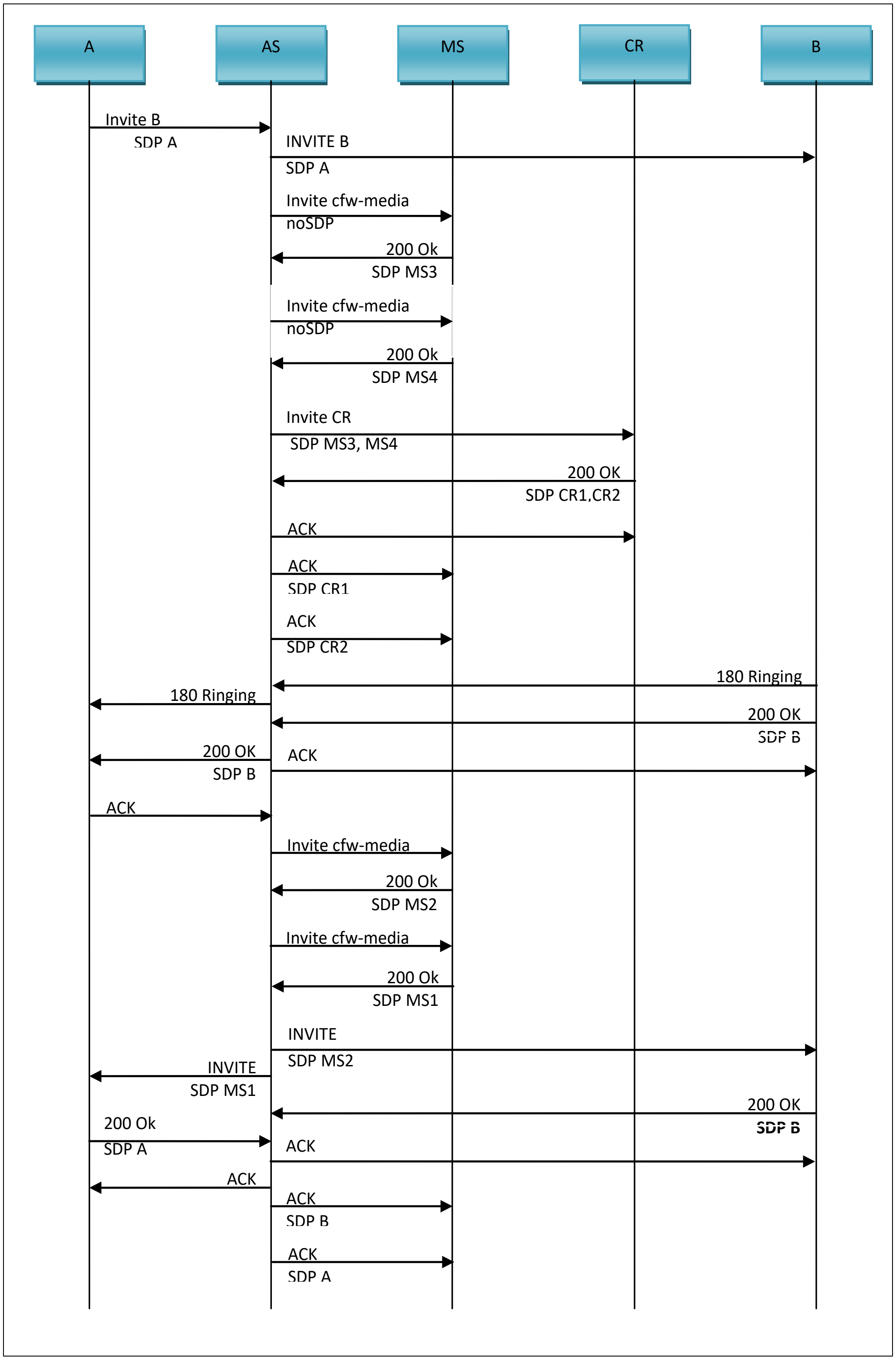

Figure 61 Terminator Recording Call

10.5.1 INVITE to Call Recording Platform

This is an example INVITE sent to the Call Recording platform. Note the inclusion of the *src* parameter in the *Contact* header and the empty *Recv-Info* header. In addition, the message body is a multipart body consisting of the SDP description and the information about the call recording is included in the Cisco BroadWorks recording metadata. Since this is a dual steam connection, there are two media lines in the SDP, one for the terminator's media and the other for the originator's media. The SDP label attributes are needed to coordinate the SDP information with the connections to the Media Server.

```
INVITE sip:3PCR@recorder.broadsoft.com:5060;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26f33f26931B5DED
From: <sip:as1.broadsoft.com>;tag=B6CB9EB1-8AE7B75C
To: sip:3PCR@recorder.broadsoft.com
Call-ID:BW153311499010488569905@10.16.134.17
CSeq:25 INVITE
Contact:<sip:as1.broadsoft.com:5060>;src
Supported:100rel
Recv-Info:
Allow:ACK,BYE,CANCEL,INFO,INVITE,OPTIONS,PRACK,REFER,NOTIFY,UPDATE
Accept:application/media_control+xml,application/sdp,multipart/mixed
Max-Forwards:10
Content-Type: multipart/mixed;boundary=deew341adf13412ferwadq
Content-Length: ...

-- deew341adf13412ferwadq
Content-Type: application/sdp

v=0
o=BroadWorks 17 1 IN IP4 10.16.134.101
s=-
c=IN IP4 10.16.134.101
t=0 0
m=audio 2896 RTP/AVP 8
a=rtpmap:8 PCMA/8000
a=sendonly
a=label:1
m=audio 2898 RTP/AVP 8
a=rtpmap:8 PCMA/8000
a=sendonly
a=label:2


-- deew341adf13412ferwadq
Content-Type: application/rs-metadata+xml
Content-Disposition: recording-session

<?xml version="1.0" encoding="UTF-8"?>
<recording-metadata xmlns="urn:ietf:params:xml:ns:recording"
id="urn:uuid:34512345-6743-6248-9043897645ab">
  <dataMode>complete</dataMode>
  <recording id="urn:uuid:894134ab-9800-7844-4456-789451125647">
    <requestor>src</requestor>
  </recording>
  <group id="urn:uuid:abc12785-4788-6654-5455-45def4522375"
         recording="urn:uuid:894134ab-9800-7844-4456-789451125647"/>
  <session id="urn:uuid:78554655-7844-5564-4568-ef4566246875"
           group="urn:uuid:abc12785-4788-6654-5455-45def4522375">
  </session>
  <stream id="urn:uuid:94dwif31-9887-341d-12id-789945621002"
          session="urn:uuid:78554655-7844-5564-4568-ef4566246875">
    <label>1</label>
```

```
    <mode>mixed</mode>
  </stream>
  <stream id="urn:uuid:34123561-7789-341d-12id-78edcaf78945"
          session="urn:uuid:78554655-7844-5564-4568-ef4566246875">
    <label>2</label>
    <mode>mixed</mode>
  </stream>
  <extensionData id="urn:uuid:ef45678456-4451-4568-7785-400554586487"
                 parent="urn:uuid:78554655-7844-5564-4568-ef4566246875">
    <broadWorksRecordingMetadata
xmlns="http://schema.broadsoft.com/broadworksCallRecording"
schemaRev="1.0"
xsi:schemaLocation="http://schema.broadsoft.com/broadworksCallRecording"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <extTrackingID>4:1</extTrackingID>
      <serviceProviderID>TxASDev</serviceProviderID>
      <groupID>North_as90</groupID>
      <userID>north02@rtx.broadsoft.com</userID>
      <callID> BW153311411129885@10.16.134.17</callID>
      <callType>
        <origCall>
          <callingPartyNumber>example@broadsoft.com</callingPartyNumber>
          <calledPartyNumber>2146415689</calledPartyNumber>
          <dialedDigits>2145551212</dialedDigits>
        </origCall>
      </callType>
      <recordingType>demand</recordingType>
    </broadWorksRecordingMetadata>
  </extensionData>
</recording-metadata>
-- deew341adf13412ferwadq—
```

10.5.2 200 OK from Call Recording Platform

This is the *200 OK* returned by the Call Recording platform. Note that it includes the *Recv-Info* header and that the *Contact* header contains the *srs* parameter. Also note that the SDP contains the two labeled media stream attributes. They are used to coordinate the connections on the Media Server.

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26f33f26931B5DED
From: <sip:as1.broadsoft.com>;tag=B6CB9EB1-8AE7B75C
To: sip:3PCR@recorder.broadsoft.com;tag=45DEF34123412DF
Call-ID: BW15331149901048856990 5@10.16.134.17
CSeq:25 INVITE
Contact:<sip:recorder.broadsoft.com:5060>;srs
Recv-Info: x-broadworks-callrecording
Allow:ACK,BYE,CANCEL,INFO,INVITE,OPTIONS,PRACK,REFER,NOTIFY,UPDATE
Content-Type: application/sdp
Content-Length: ...

v=0
o=BroadWorks 17 1 IN IP4 10.16.134.110
s=-
c=IN IP4 10.16.134.110
t=0 0
m=audio 26680 RTP/AVP 8
a=rtpmap:8 PCMA/8000
a=recvonly
a=label:1
m=audio 26682 RTP/AVP 8
```

```
a=rtpmap:8 PCMA/8000
a=recvonly
a=label:2
```

10.5.3 ACK to Call Recording Platform

This is an example of the ACK sent to the Call Recording platform. Note the inclusion of the *src* parameter in the *Contact* header.

```
ACK sip:recorder.broadsoft.com SIP/2.0
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26ff4f268799455
From: <sip:as1.broadsoft.com>;tag= B6CB9EB1-8AE7B75C
To: sip:3PCR@recorder.broadsoft.com;tag=45DEF34123412DF
Call-ID: BW153311499010488569905@10.16.134.17
CSeq:25 ACK
Contact:<sip:as1.broadsoft.com:5060>;src
Content-Length: 0
```

10.6 Feature Access Code for On-Demand Notification

This flow shows an example of the INFO message sent to the Call Recording platform for the call setup in the previous section. That call was set up in *On-Demand* mode.

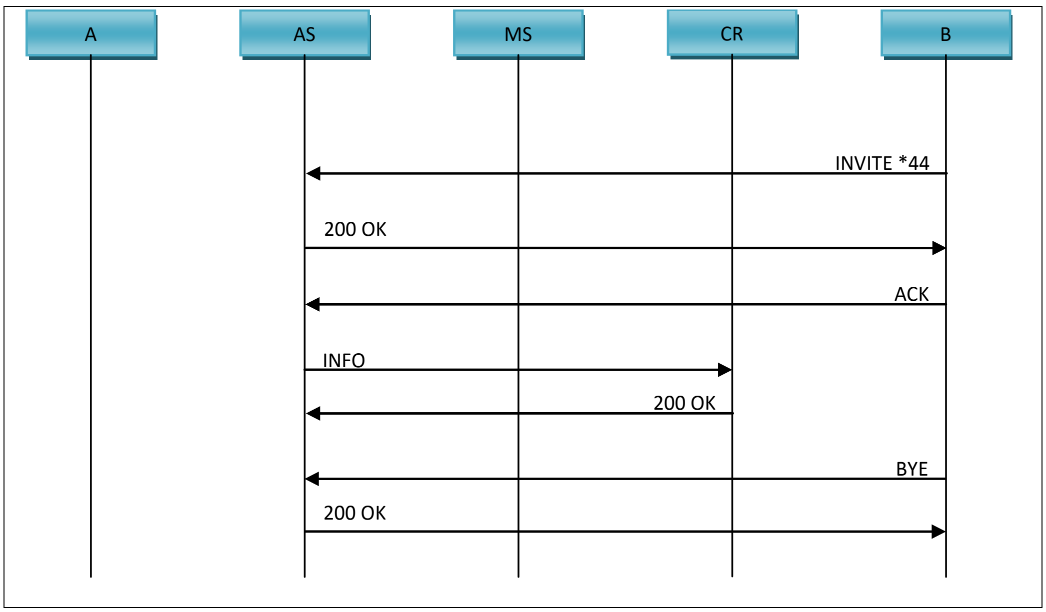

Figure 62 FAC for On-Demand Notification

10.6.1 INFO Message

This is an example of the INFO message sent by the Application Server to the Call Recording platform. There is no message body in the INFO. The presence of the *Info-Package* and the fact that the INFO message is part of the ongoing dialog for the call recording, provide sufficient information to identify the call recording to be kept. The Application Server does not send a message body in the INFO.

```
INFO sip:recorder.broadsoft.com SIP/2.0
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26ff4f44887554
From: <sip:as1.broadsoft.com>;tag= B6CB9EB1-8AE7B75C
To: sip:3PCR@recorder.broadsoft.com;tag=45DEF34123412DF
```

```
Call-ID: BW153311499010488569905@10.16.134.17
CSeq:26 INFO
Contact:<sip:as1.broadsoft.com:5060>;src
Max-Forwards: 70
Info-Package: x-broadworks-callrecording
Content-Length: 0
```

10.7 User Redirects Call

In this example, the originator transfers the call to another party. The originator is recording the call prior to the transfer. The update is sent to inform the 3PCR platform that the call is being transferred, and the recording is released when the transfer succeeds.

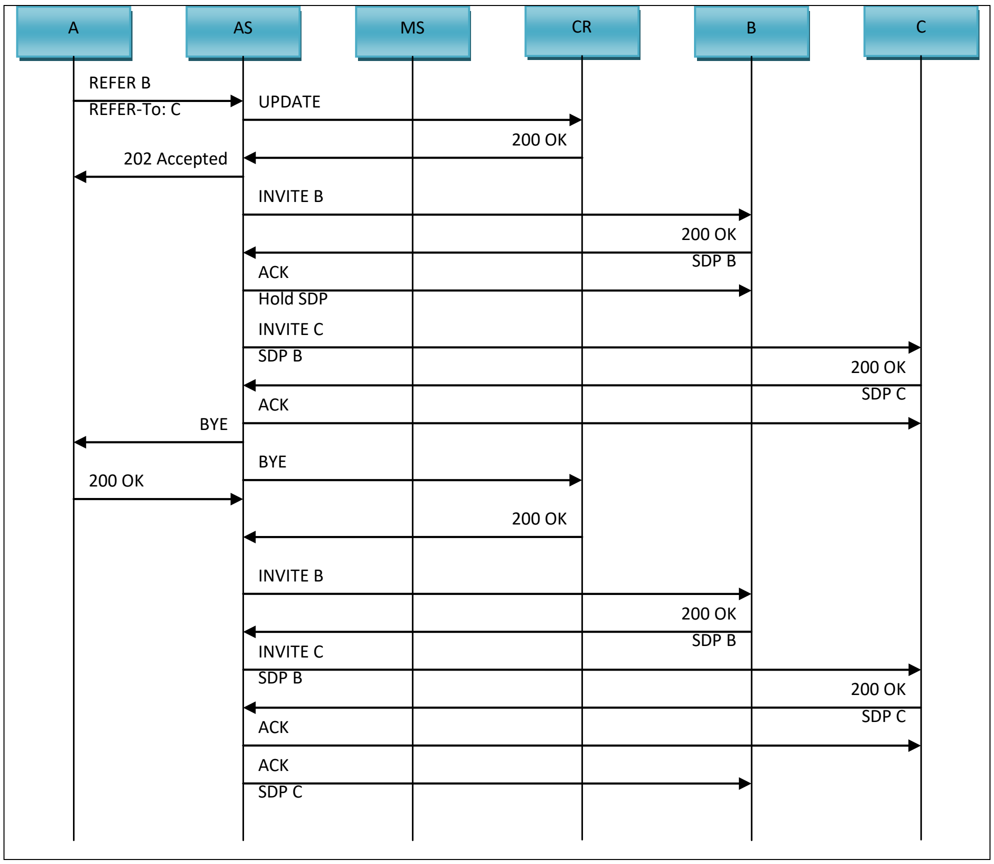

Figure 63 User Redirects Call

10.7.1 UPDATE Example

This is an example of the UPDATE message sent by the Application Server. In this case, it contains the redirect information about the call transfer.

```
UPDATE sip:3PCR@recorder.broadsoft.com:5060;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.16.134.101;branch=z9hG4bK26f33f26931B5DED
From: <sip:as1.broadsoft.com>;tag=B6CB9EB1-8AE7B75C
To: sip:3PCR@recorder.broadsoft.com;tag=CDASE-123134ad
Call-ID:BW153311499010488569905@10.16.134.17
CSeq:29 UPDATE
Contact:<sip:as1.broadsoft.com:5060>;src
```

```
Supported:100rel
Allow:ACK,BYE,CANCEL,INFO,INVITE,OPTIONS,PRACK,REFER,NOTIFY,UPDATE
Accept:application/media_control+xml,application/sdp,multipart/mixed
Max-Forwards:10
Content-Type: application/rs-metadata+xml
Content-Disposition: recording-session
Content-Length: ...

<?xml version="1.0" encoding="UTF-8"?>
<recording-metadata xmlns="urn:ietf:params:xml:ns:recording"
id="urn:uuid:34512345-6743-6248-9043897645ab">
  <dataMode>partial</dataMode>
  <extensionData id="urn:uuid:ef45678456-4451-4568-7785-400554586487"
                 parent="urn:uuid:78554655-7844-5564-4568-ef4566246875">
    <broadWorksRecordingMetadata
xmlns="http://schema.broadsoft.com/broadworksCallRecording"
schemaRev="1.0"
xsi:schemaLocation="http://schema.broadsoft.com/broadworksCallRecording"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <extTrackingID>4:1</extTrackingID>
      <redirectCallInfo>
        <newExtTrackingID>7:1</newExtTrackingID>
        <redirectedFromPartyNumber>9725551212@broadsoft.com
        </redirectedFromPartyNumber>
        <redirectedToPartyNumber>9725551414@broadsoft.com
        </redirectedToPartyNumber>
        <\redirectCallInfo>
    </broadWorksRecordingMetadata>
  </extensionData>
</recording-metadata>
```

10.8 Controls for IP Phones Message Call Flows

The following call flows do not cover the messaging that occurs between the Cisco BroadWorks Application Server and the Media Server or the 3PCR platform. For those message call flows, see the *Call Recording – Start, Stop, Pause, Resume User Control Feature Description* [12].

For the sake of brevity, the primary focus of these call flows is the messaging between the recording-aware UA and the Cisco BroadWorks Application Server. Even though the Media Server and the 3PCR platform are depicted, they are merely present in the call flows to indicate they are involved in the recording.

10.8.1 Subscribe to Device Feature Synchronization Event Package

If the call recording user's SIP endpoint device is subscribed to the Device Feature Synchronization event package, the new *CallRecordingModeEvent* is sent in the body of the SIP NOTIFY, and it indicates the call recording user's currently provisioned recording mode.

Figure 64 shows a typical sequence in which a Cisco BroadWorks user is registering and subscribing for the Device Feature Synchronization event package (*as-feature-event*). In the initial subscription notification, the NOTIFY body may include any relevant Device Feature Synchronization events for the Cisco BroadWorks user, such as the *CallRecordingModeEvent*.

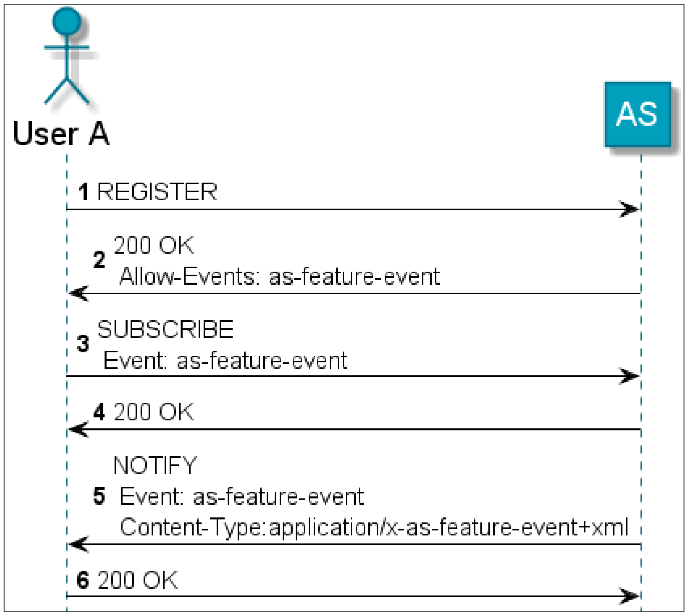

Figure 64 REGISTER and SUBSCRIBE for Device Feature Synchronization Event Package

If the *CallRecordingModeEvent* is present in the body of the NOTIFY, it conveys the user's provisioned recording mode. For an example of the NOTIFY body content, see section *9.7.5 Example NOTIFY with CallRecordingModeEvent*.

10.8.2 Mid-Call Provisioning Changes to Recording Mode

The call recording user's provisioned recording mode may be modified from Cisco BroadWorks through the web portal or the Xtended Services Interface.

If the user's recording mode is modified after the initial subscription notification is sent to the SIP endpoint device, the newly modified recording mode is indicated in the *CallRecordingModeEvent* of a subsequent NOTIFY, as shown in *Figure 65*.

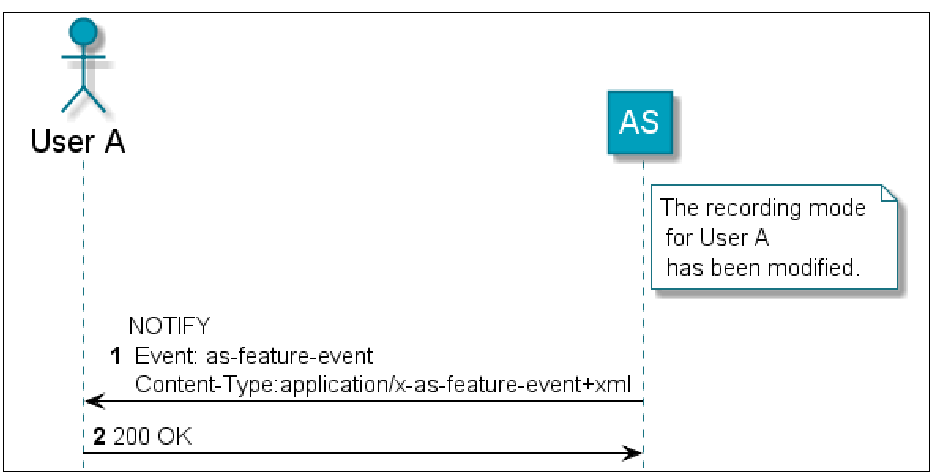

Figure 65 NOTIFY Sent When Provisioned Recording Mode Modified

It is possible that the user's recording mode is modified while the user has an active call (or active calls). During mid-call provisioning changes to a user's recording mode, the newly provisioned recording mode does **not** take effect on the Cisco BroadWorks Application Server for the user's active call(s). The user's active call(s) continue to use the recording mode that was provisioned at the time the call was established. The newly provisioned recording mode only takes effect for the user's subsequent calls.

NOTE 1 (on recording-aware client behavior):

When the provisioned recording mode is modified, the Cisco BroadWorks Application Server sends a NOTIFY to the recording-aware UA that contains the user's newly modified recording mode. The NOTIFY may arrive while the user is still on an active call (or active calls), in which case, the recording-aware UA must continue to display the appropriate recording controls based on the original recording mode at the time the call was established. For the user's subsequent new calls, the recording-aware UA must display the appropriate controls based on the newly modified recording mode.

NOTE 2 (on recording-aware client behavior):

There is currently no mechanism for the recording-aware device to modify the recording mode via the SIP interface.

10.8.3 Originating Party – Call Recording Begins Automatically

In certain call recording modes, such as *Always* or *Always with Pause/Resume Support*, the recording is set up automatically when the user initiates or receives a call.

In *Figure 66,* User A is the originating party assigned with the Call Recording service in *Always* or *Always with Pause/Resume Support* recording mode. When User A initiates the call, the INVITE sent by User A includes the *Supported* header with the *record-aware* option tag. The Cisco BroadWorks Application Server recognizes that User A is recording aware. Therefore, when the call recording is underway, the Cisco BroadWorks Application Server sets the *a=record* attribute in the SDP to "on".

NOTE (on recording-aware client behavior):

When the recording begins for the user in *Always* recording mode, the Cisco BroadWorks Application Server notifies the recording-aware UA of the recording state with the *a=record* attribute in the SDP set to "on". Based on the subscription's NOTIFY that contains the *CallRecordingModeEvent* and the *Always* recording mode, the recording-aware device should realize that it should **not** display any recording controls on the phone's display when the user is on an active call that is recording in *Always* mode.

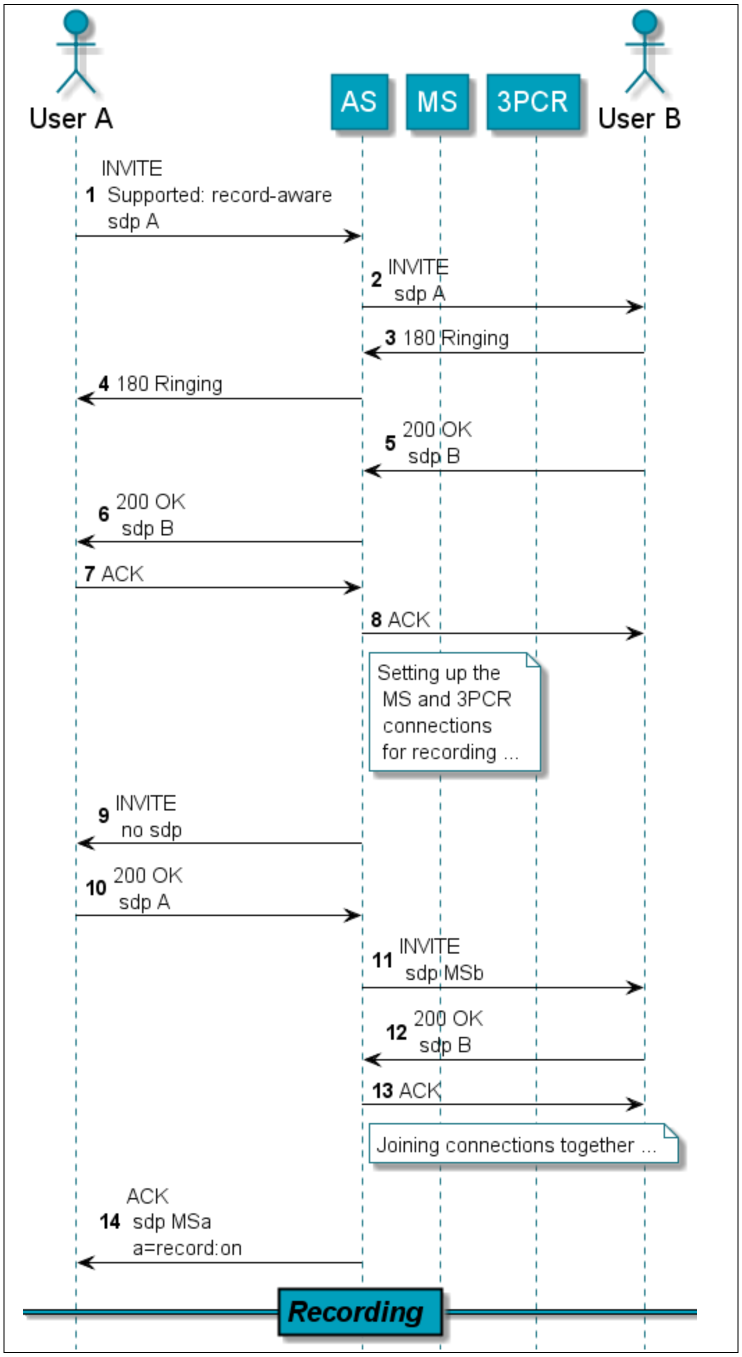

Figure 66 Originating Party – Call Recording Begins Automatically

10.8.4 Terminating Party – Call Recording Begins Automatically

Similarly, in *Figure 67*, if User A is the terminating party assigned with the Call Recording service in *Always* or *Always with Pause/Resume Support* recording mode, when User A responds to the initial INVITE sent by User B, User A includes the *Supported* header with the *record-aware* option tag. The Cisco BroadWorks Application Server recognizes that User A is recording aware, so when the call recording is underway, the Cisco BroadWorks Application Server sets the *a=record* attribute in the SDP to "on".

NOTE (on recording-aware client behavior):

When the recording begins for the user in *Always* recording mode, the Cisco BroadWorks Application Server notifies the recording-aware UA of the recording state with the *a=record* attribute in the SDP set to "on". Based on the subscription's NOTIFY that contains the *CallRecordingModeEvent* and the *Always* recording mode, the recording-aware device should realize that it should **not** display any recording controls on the phone's display when the user is on an active call that is recording in *Always* mode.

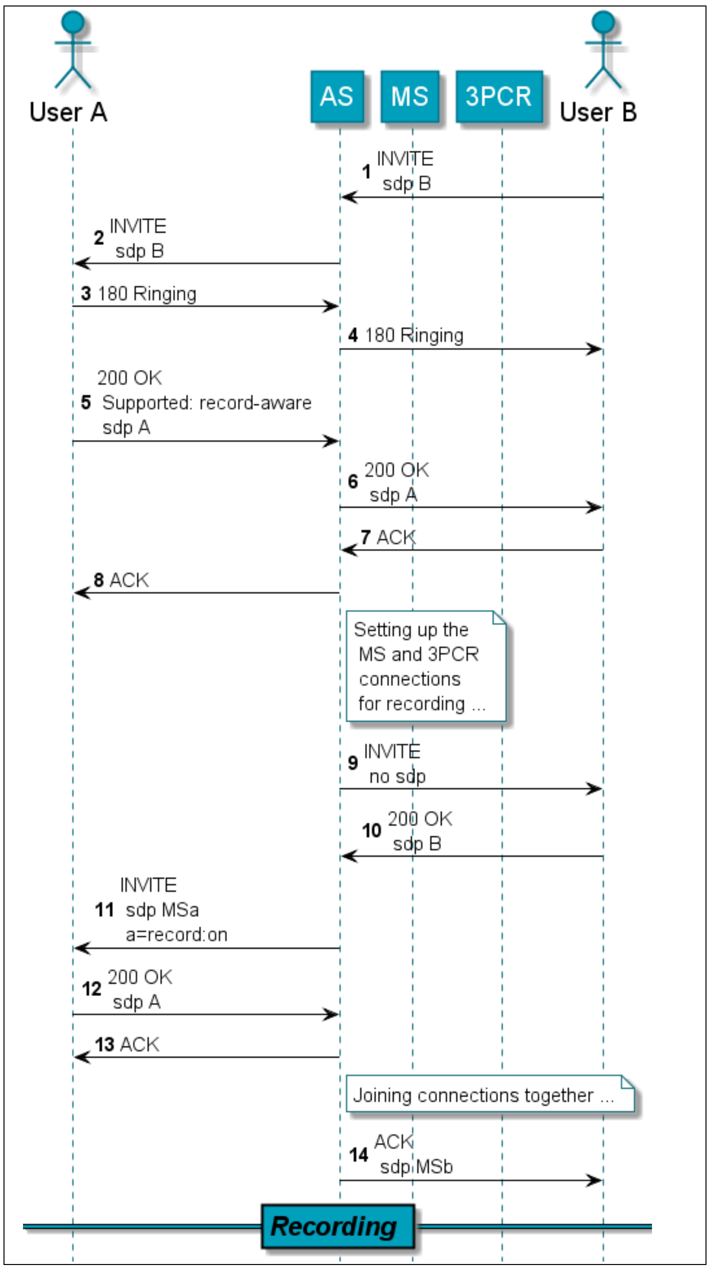

Figure 67 Terminating Party – Call Recording Begins Automatically

10.8.5 Both Parties – Call Recordings Begin Automatically

Figure 68 and *Figure 69* show the message flow when call recording begins automatically for both the originating and the terminating parties.

In this example, User A is the originating party and User B is the terminating party. Both have been assigned the Call Recording service in *Always* or *Always with Pause/Resume Support* recording mode. User A initiates the call to the Cisco BroadWorks Application Server, and the INVITE sent by User A includes the *Supported* header with the *record-aware* option tag. When the Cisco BroadWorks Application Server terminates this call to User B, User B also sends the *Supported* header with the *record-aware* option tag to the Cisco BroadWorks Application Server.

The Cisco BroadWorks Application Server recognizes that both parties are recording aware. As the two call recordings are being established, the Cisco BroadWorks Application Server sets the *a=record* attribute in the SDPs to "on" accordingly.

NOTE (on recording-aware client behavior):

When the recording(s) begin(s) for the user(s) in *Always* recording mode, the Cisco BroadWorks Application Server notifies the recording-aware UA(s) of the recording state with the *a=record* attribute in the SDP set to "on". Based on the subscription's NOTIFY that contains the *CallRecordingModeEvent* and the *Always* recording mode, the recording-aware device(s) should realize that it should **not** display any recording controls on the phone's display when the user(s) is on an active call that is recording in *Always* mode.

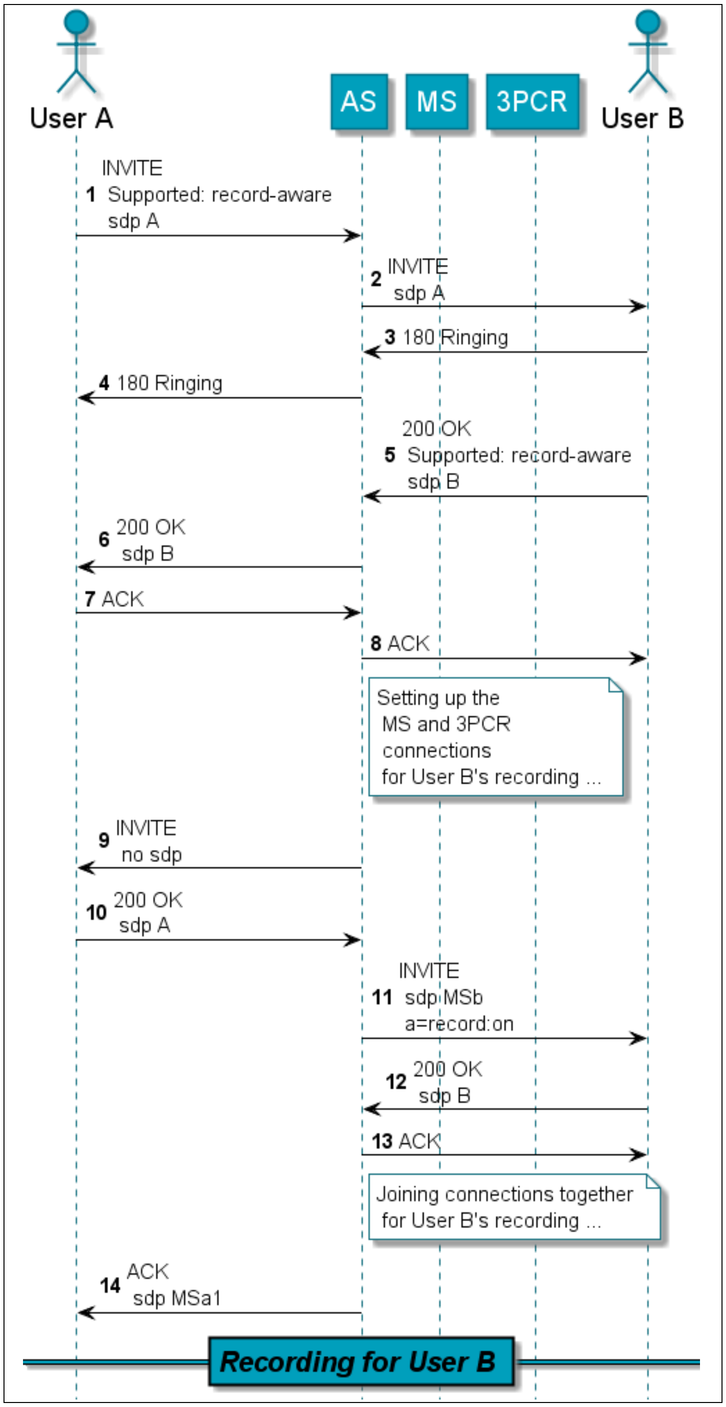

Figure 68 Both Parties – Call Recordings Begin Automatically (a)

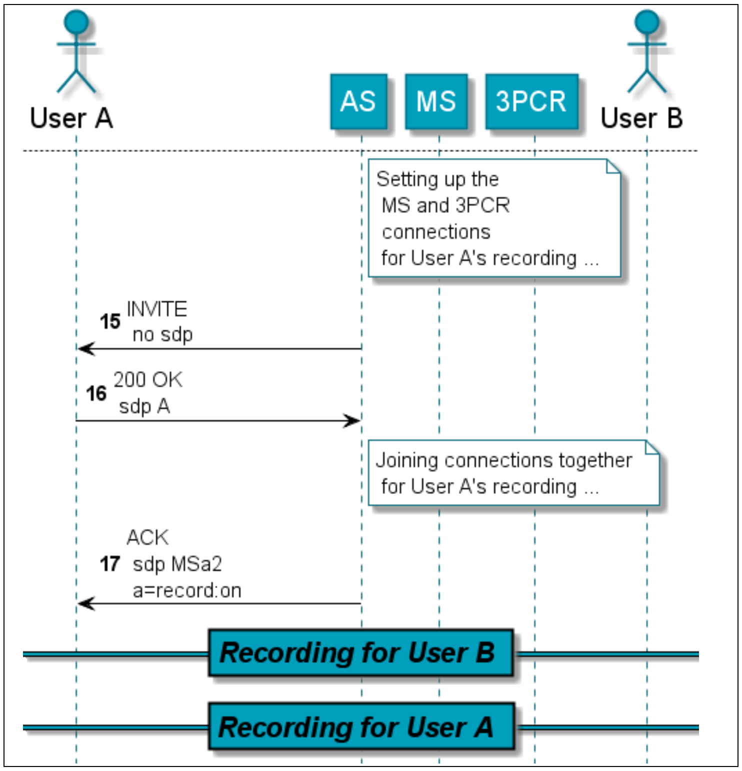

Figure 69 Both Parties – Call Recordings Begin Automatically (b)

10.8.6 Originating Party Starts Mid-Call Recording on Active Call

In the *On Demand with User Initiated Start* recording mode, the recording does not start automatically. The user must start the recording manually, as shown in *Figure 70.*

When User A initiates the call, the INVITE sent by User A includes the *Supported* header with the "record-aware" option tag. The Cisco BroadWorks Application Server recognizes that User A is recording aware. Since there is no recording until the user starts one, the Cisco BroadWorks Application Server sets the *a=record* attribute in the SDP to "off".

To start the recording, User A can send an INVITE with the *a=recordpref* attribute in the SDP set to "on". This triggers the Call Recording service to start the recording. When the recording is underway, the Cisco BroadWorks Application Server sets the *a=record* attribute in the SDP to "on".

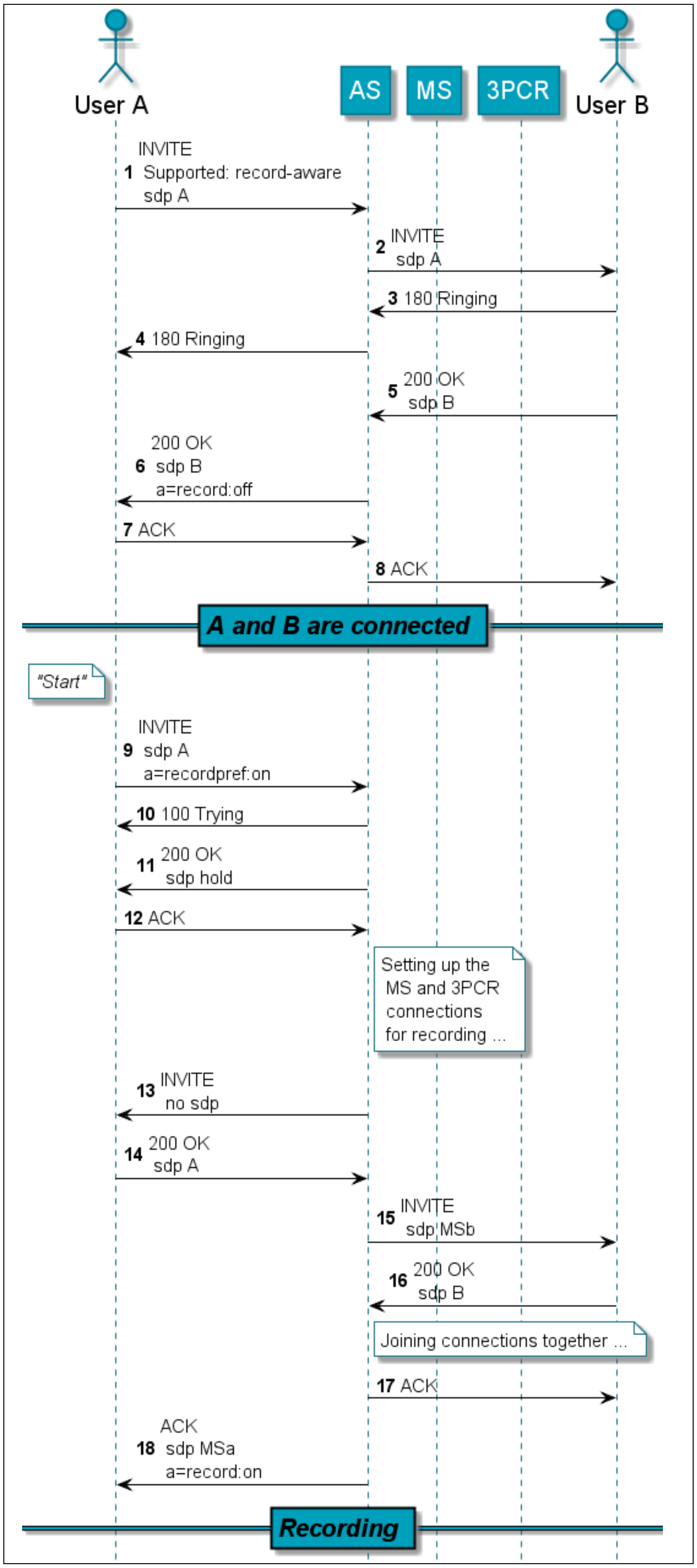

Figure 70 Originating Party Starts Mid-Call Recording on Active Call

10.8.7 Originating Party Starts Mid-Call Recording After Placing Call On Hold

If the *On Demand with User Initiated Start* user wants to start recording a call that the user has placed on hold, the *a:recordpref=on* attribute cannot be processed until the call is taken off of hold first. When the call is taken off of hold, the Cisco BroadWorks Application Server can establish the recording connections and then send the *a=record* attribute set to "on" when recording has begun.

This scenario is shown in *Figure 71* and *Figure 72*. User A, with *On Demand with User Initiated Start* mode, initiates a call to User B. The INVITE sent by User A includes the *Supported* header with the *record-aware* option tag. The Cisco BroadWorks Application Server recognizes that User A is recording aware. Since there is no recording until the user starts one, the Cisco BroadWorks Application Server sets the *a=record* attribute in the SDP to "off". Next, User A places User B on hold and then attempts to start a recording by sending an INVITE with the *a=recordpref* attribute in the SDP set to "on". Because the call to User B is currently on hold, the Call Recording service cannot start the recording yet. When User A takes User B off of hold, this triggers the Call Recording service to establish the necessary connections for the recording. When the recording is underway, the Cisco BroadWorks Application Server sets the *a=record* attribute in the SDP to "on".

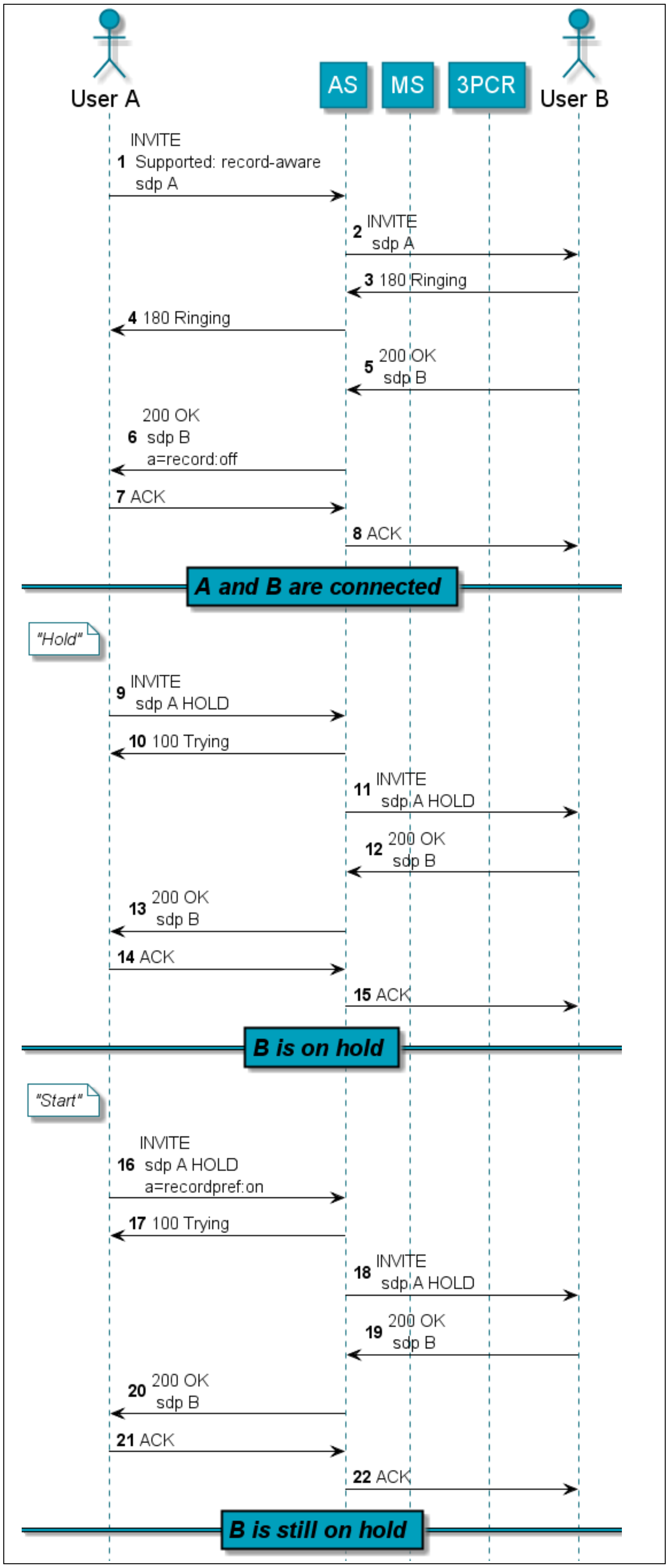

Figure 71 Originating Party Starts Mid-Call Recording After Placing Call On Hold (a)

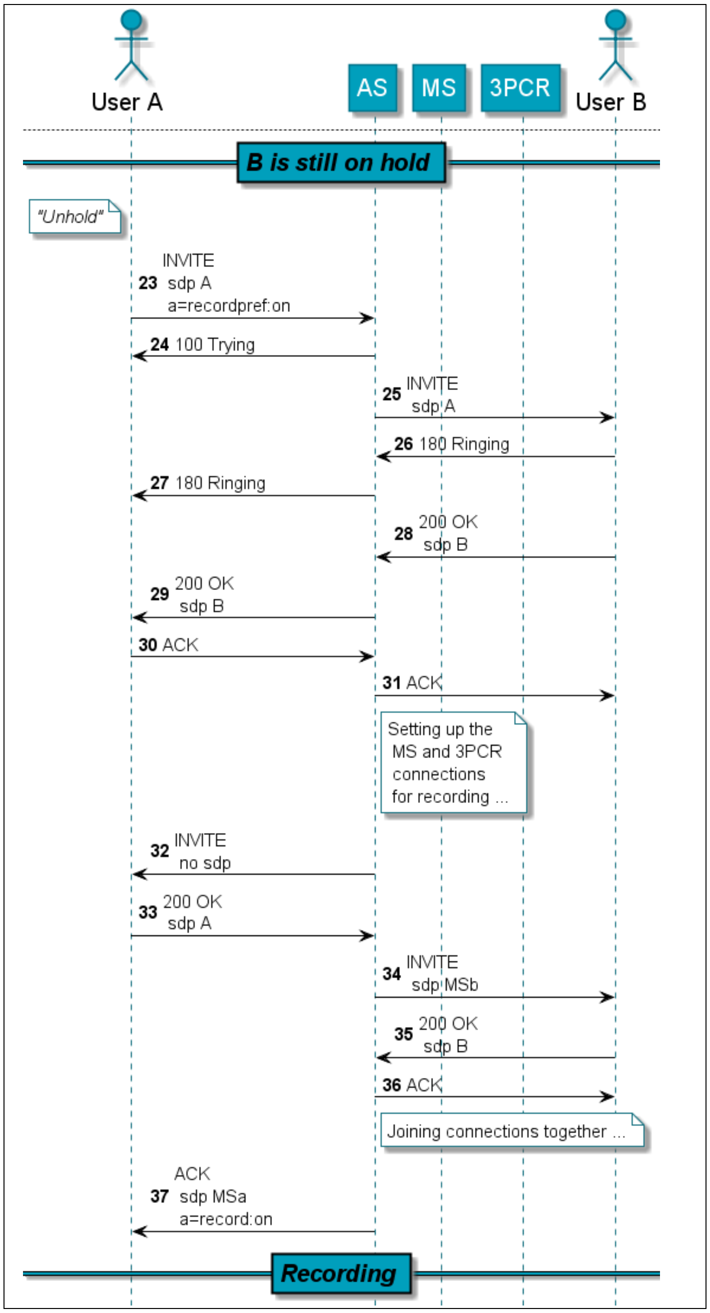

Figure 72 Originating Party Starts Mid-Call Recording After Placing Call On Hold (b)

10.8.8 Terminating Party Starts Mid-Call Recording on Active Call (re-INVITE)

In *Figure 73*, if the user with *On Demand with User Initiated Start* mode is the terminating party, the recording does not start automatically. The user must start the recording manually.

When User A responds to the INVITE sent by User B, User A includes the *Supported* header with the *record-aware* option tag. The Cisco BroadWorks Application Server recognizes that the user is recording aware, but when the call recording user is the terminating party, the Cisco BroadWorks Application Server does **not** send the *a=record* attribute in the SDP, since the offer-answer is completed. For more information, see section *4.2.5.3.2 record*.

To start the recording, User A can send a re-INVITE with the *a=recordpref* attribute in the SDP set to "on". This triggers the Call Recording service to start the recording. When the recording is underway, the Cisco BroadWorks Application Server sets the *a=record* attribute in the SDP to "on".

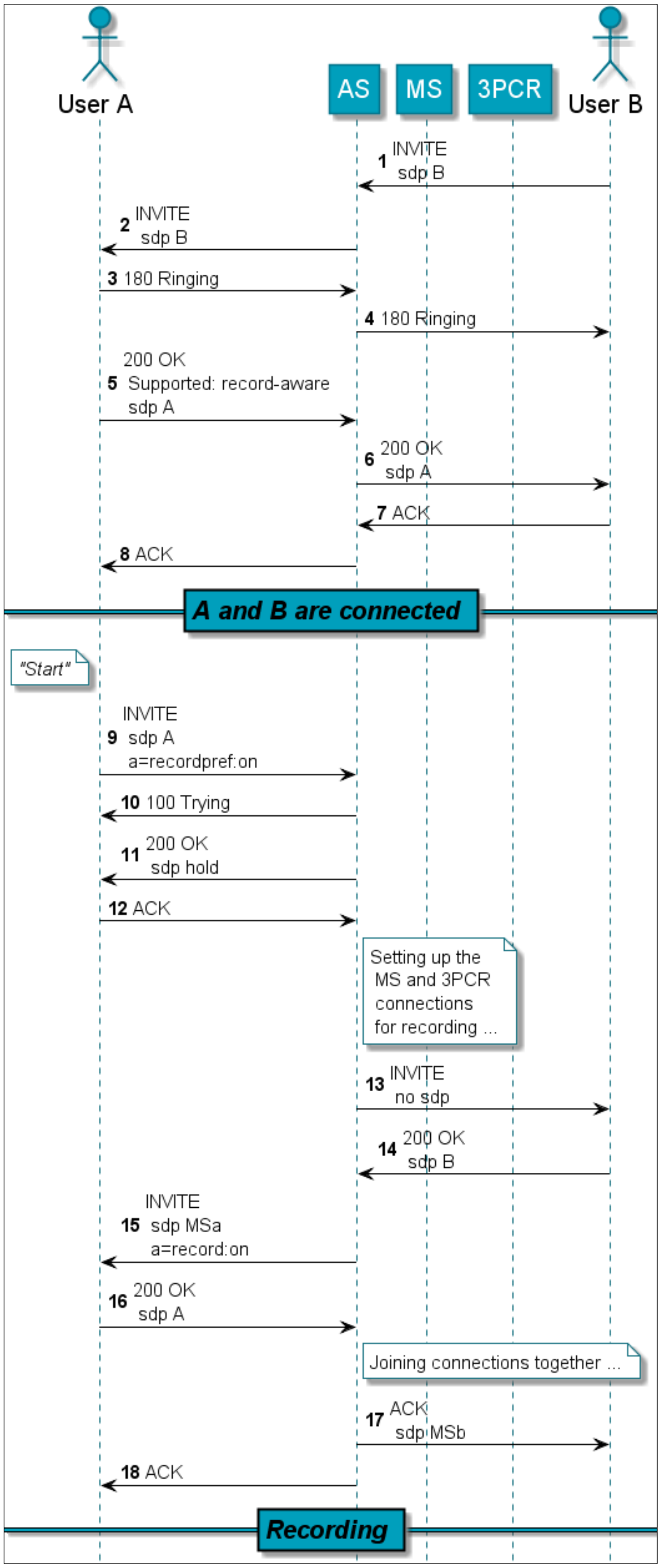

Figure 73 Terminating Party Starts Mid-Call Recording on Active Call (re-INVITE)

10.8.9 Terminating Party Starts Mid-Call Recording on Active Call (UPDATE)

Figure 74 shows how the recording-aware UA can send an UPDATE with the *a=recordpref* attribute set to “on”, instead of sending it in a re-INVITE. The following is the same call flow depicted in *Figure 73*, but with User A’s re-INVITE message sequence changed to an UPDATE message sequence. The UPDATE with *a=recordpref* set to “on” triggers the Call Recording service to start the recording. When the recording is underway, the Cisco BroadWorks Application Server sets the *a=record* attribute in the SDP to “on”.

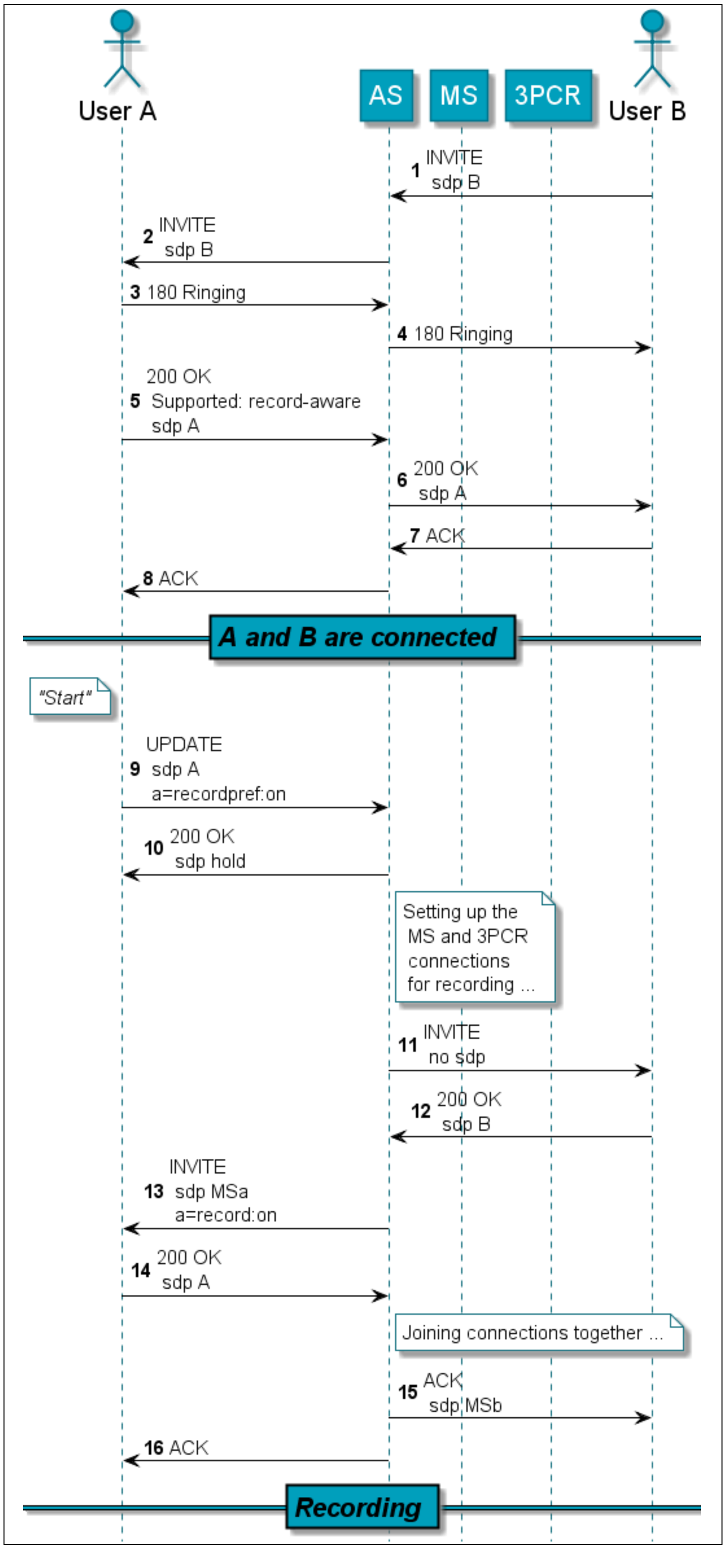

Figure 74 Terminating Party Starts Mid-Call Recording on Active Call (UPDATE)

10.8.10 Terminating Party Starts Mid-Call Recording After Placing Call On Hold

If the *On Demand with User Initiated Start* user wants to start recording a call that the user has placed on hold, the *a:recordpref=on* attribute cannot be processed until the call is taken off of hold first. When the call is taken off of hold, the Cisco BroadWorks Application Server can establish the recording connections and then send the *a=record* attribute set to "on" when recording has begun.

This scenario is shown in *Figure 75* and *Figure 76*. This is similar to *Figure 71* and *Figure 72*, with slight messaging differences because User A is the terminating party.

In the following figures User B initiates a call to User A, who has *On Demand with User Initiated Start* mode. The Cisco BroadWorks Application Server recognizes that User A is recording aware when it sees the *Supported* header with the *record-aware* option tag sent by User A; however, when the call recording user is the terminating party, the Cisco BroadWorks Application Server does not send the *a=record* attribute in the SDP, since the offer-answer is completed. For more information, see section *4.2.5.3.2 record*. Next, User A places User B on hold, and then attempts to start a recording by sending an INVITE with the *a=recordpref* attribute in the SDP set to "on". Since the call to User B is currently on hold, the Call Recording service cannot start the recording yet. When User A takes User B off of hold, this triggers the Call Recording service to establish the necessary connections for the recording. When the recording is underway, the Cisco BroadWorks Application Server sets the *a=record* attribute in the SDP to "on".

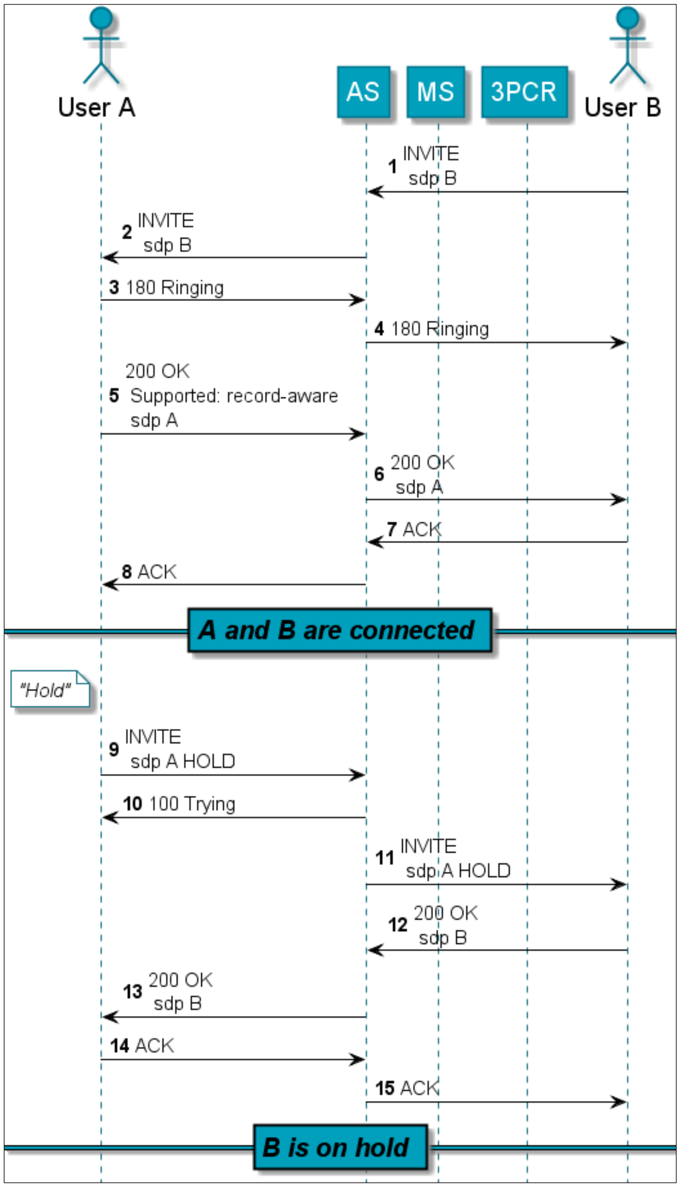

Figure 75 Terminating Party Starts Mid-Call Recording After Placing Call On Hold (a)

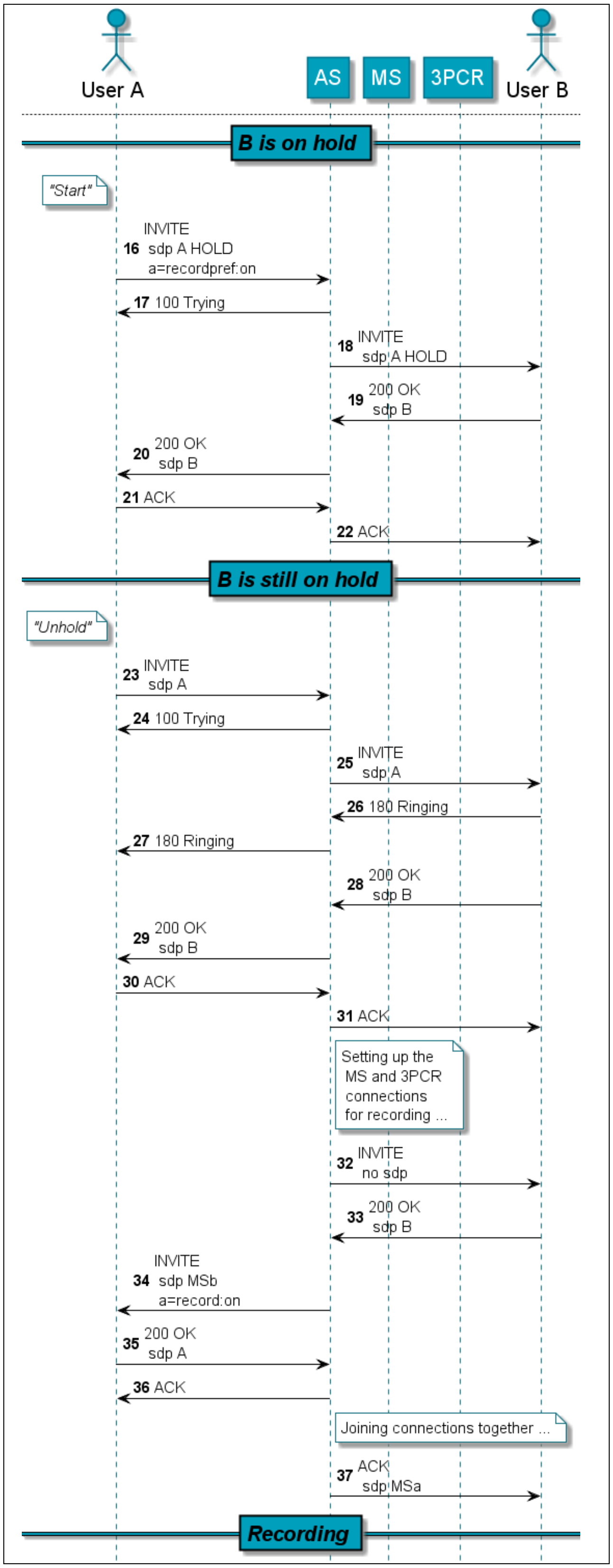

Figure 76 Terminating Party Starts Mid-Call Recording After Placing Call On Hold (b)

10.8.11 Originating Party Starts Call Recording During Call Setup

For the recording modes, *On Demand* or *On Demand with User Initiated Start*, the user may want to initiate the recording at the same time as the user initiates the call. The Cisco BroadWorks Application Server can support receiving the *a=recordpref* attribute set to "on" in the initial INVITE during call setup, as shown in *Figure 77*. Upon successfully establishing the recording connections, the Cisco BroadWorks Application Server recognizes the recording as saved and the call flow resembles *Figure 66*, where the *a=record* attribute is set to "on".

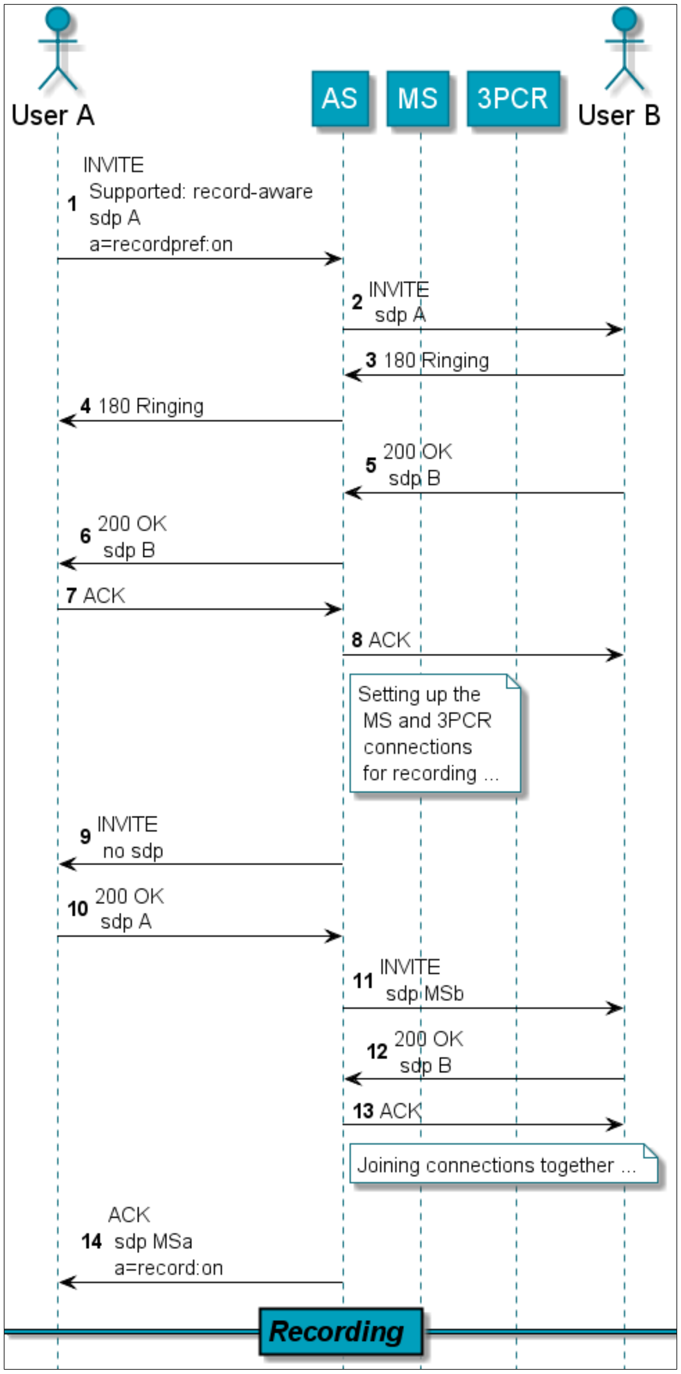

Figure 77 Originating Party Starts Call Recording During Call Setup

10.8.12 Mid-Call Pause Recording and Mid-Call Resume Recording

Certain recording modes allow the user to pause and resume the recording. These modes include *Always with Pause/Resume Support*, *On Demand*, and *On Demand with User Initiated Start.*

In *Figure 78*, User A has been assigned the Call Recording service with recording mode *Always with Pause/Resume Support*, *On Demand*, or *On Demand with User Initiated Start*. To pause, User A requests the recording to be paused by setting the *a=recordpref* attribute in the SDP to "pause". Once the recording has been paused, the Cisco BroadWorks Application Server sends the *a=record* attribute with "paused" in the SDP.

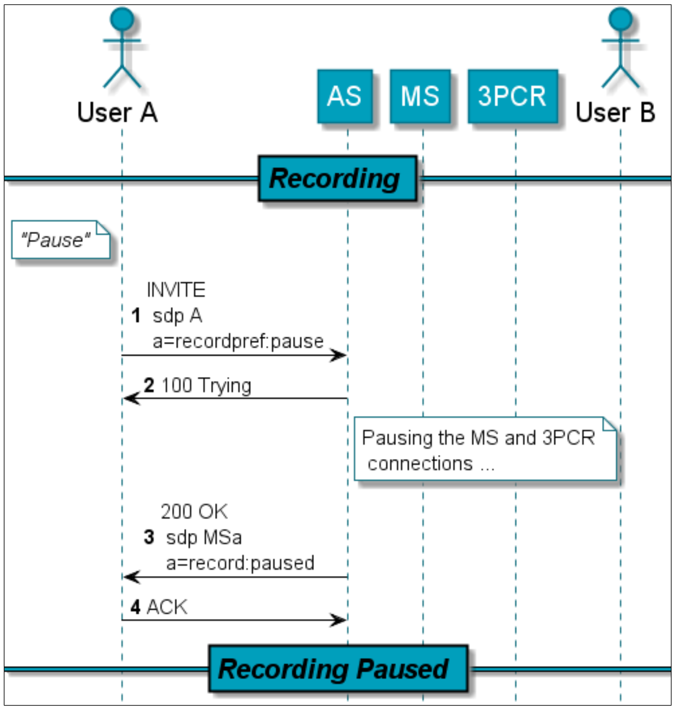

Figure 78 Mid-Call Pause Recording

Similarly, for resume in *Figure 79*, User A requests the recording to be resumed by setting the *a=recordpref* attribute in the SDP to "on". Once the recording has been resumed, the Cisco BroadWorks Application Server sends the *a=record* attribute with "on" in the SDP.

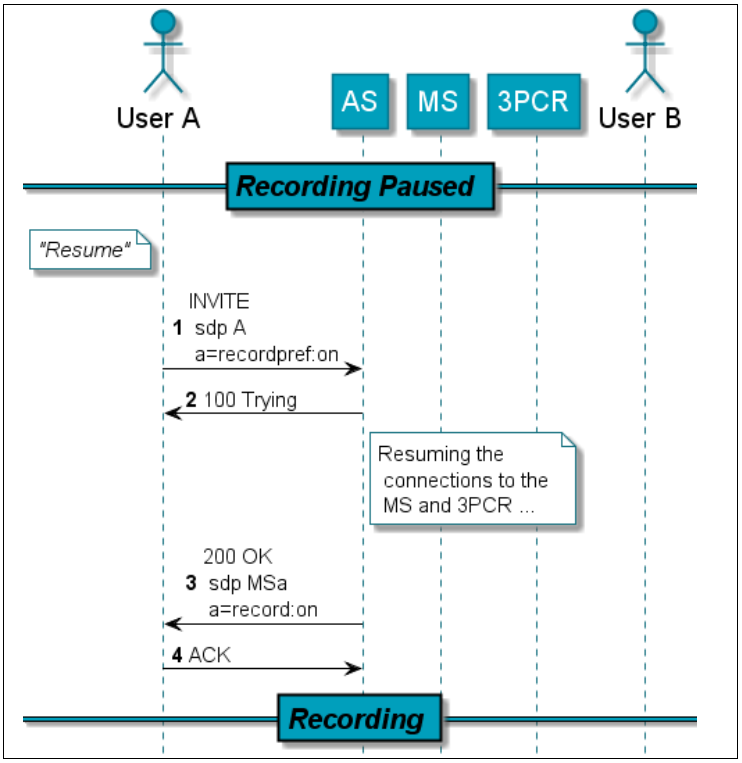

Figure 79 Mid-Call Resume Recording

10.8.13 Mid-Call Stop Recording

The only recording mode that allows the call recording user to stop the recording is *On Demand with User Initiated Start*.

In *Figure 80*, User A has been assigned the Call Recording service with recording mode *On Demand with User Initiated Start*. To stop the recording, User A requests the recording to be stopped by setting the *a=recordpref* attribute in the SDP to "off". Once the recording has been stopped, the Cisco BroadWorks Application Server sends the *a=record* attribute with "off" in the SDP to User A. User A and User B are reconnected when the Media Server and 3PCR platform are no longer involved in the media path for recording.

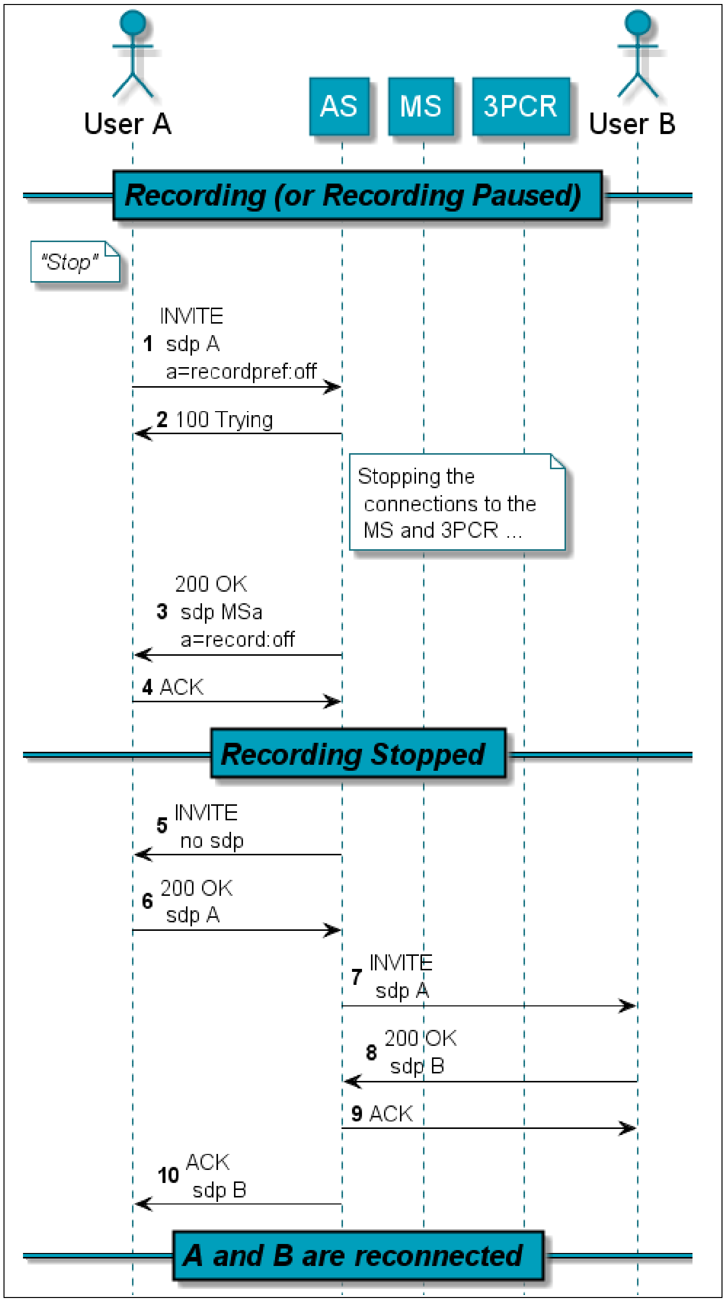

Figure 80 Mid-Call Stop Recording

10.8.14 On Demand Recording Mode – Save Recording

Note that if the user has the *On Demand* recording mode, the recording starts automatically (as is the case with *Always* and *Always with Pause/Resume Support* modes); however, the user must manually request that the recording be saved. This scenario is shown in *Figure 81*.

> **NOTE (on recording-aware client behavior)**:
>
> To differentiate between the *On Demand* mode and the other modes that start recording automatically, the Cisco BroadWorks Application Server sets the *a=record* attribute to "off" if the user has the *On Demand* recording mode. Even though the recording has already started on behalf of the user, the "off" value indicates to the recording-aware UA that it should display a control to allow the user to start (that is, "save") the recording.

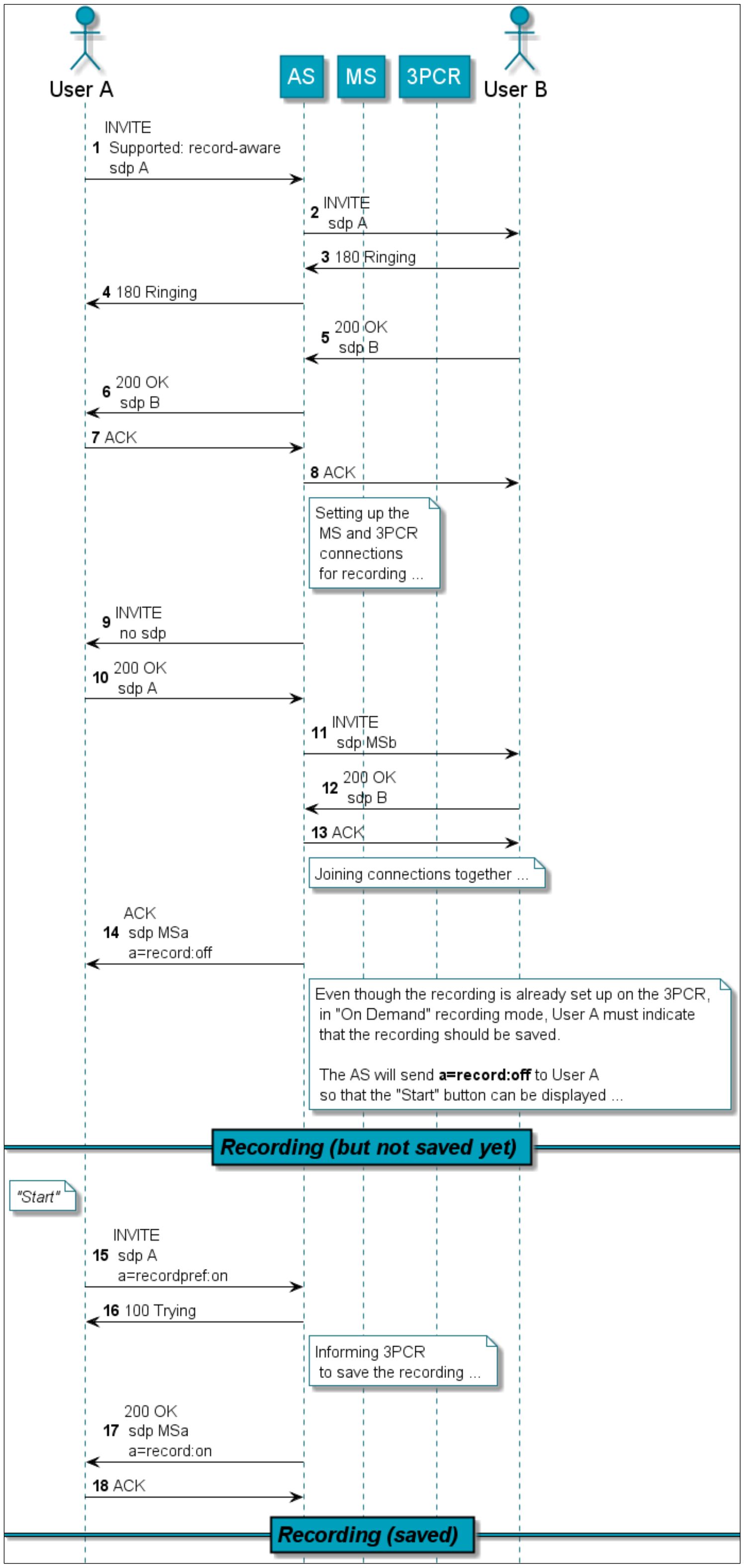

Figure 81 On Demand Recording Mode – Save Recording

10.8.15 Never Recording Mode

Note that if the user is assigned the *Never* recording mode, call recording is never initiated on the Cisco BroadWorks Application Server for the user's calls. The messaging from the Cisco BroadWorks Application Server to the recording-aware UA during call setup does not contain the *a=record* attribute set to "off" in the SDP to indicate there is no recording.

> **NOTE (on recording-aware client behavior)**:
>
> When the recording-aware device receives the subscription's NOTIFY that contains the *CallRecordingModeEvent* and the *Never* recording mode, the recording-aware device should realize that the user's calls are **not** to be recorded. Even though the Cisco BroadWorks Application Server does not send the *a=record* attribute set to "off", the recording-aware device should already understand that the calls for this user are **not** recorded and that recording actions are **not** to be displayed on the phone.

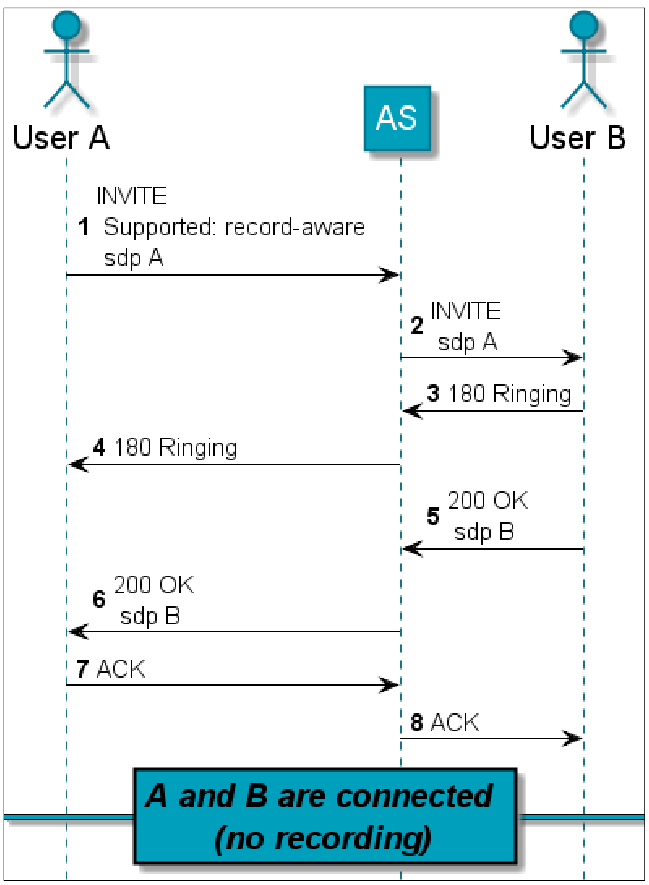

Figure 82 Never Recording Mode

10.8.16 Xtended Services Interface Requests Resulting in Recording Changes

It is possible for the state of the recording to be altered externally by the user from another interface other than the SIP endpoint device. Requests via the Xtended Services Interface can be issued to start, pause, resume, and/or stop a call recording. If this occurs, the Cisco BroadWorks Application Server informs the recording-aware UA of the recording's altered state, and the display can be updated accordingly.

The following figures depict start, pause, resume, and stop requests via the call recording user's Xtended Services Interface client, which results in the update to the user's recording-aware IP phone.

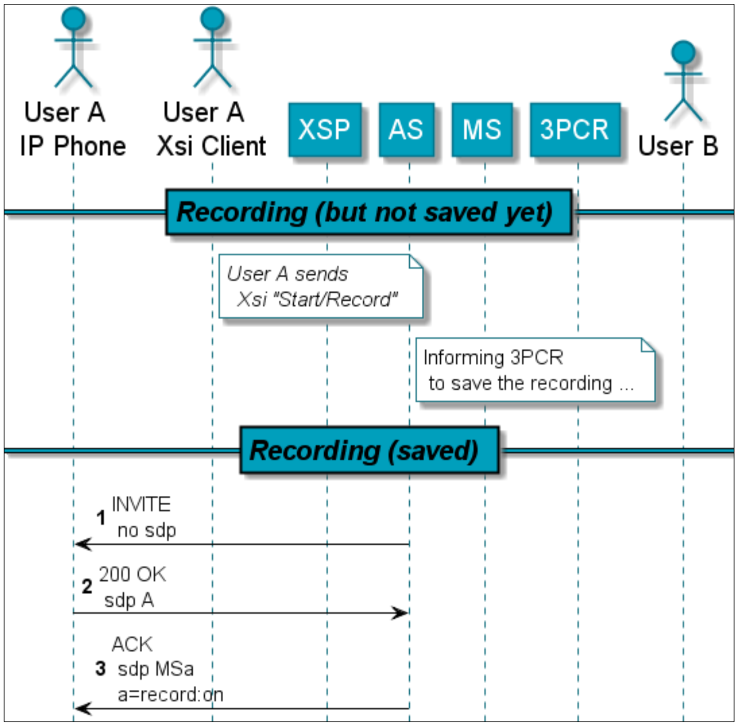

Figure 83 Xtended Services Interface Start Recording (On Demand)

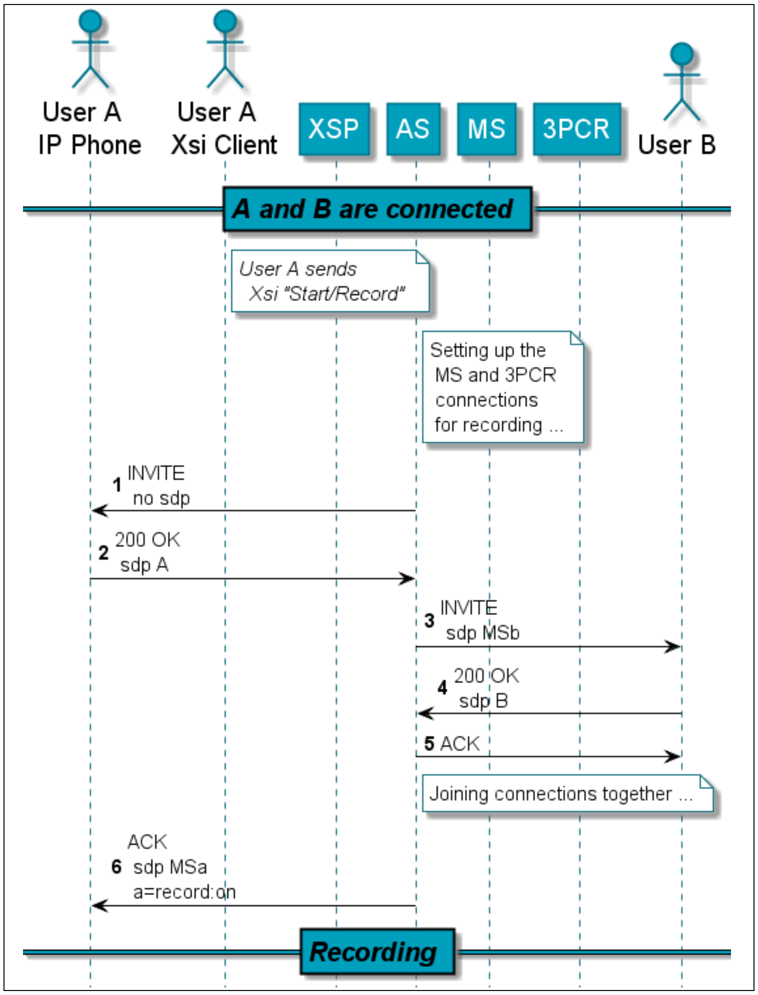

Figure 84 Xtended Services Interface Start Recording (On Demand with User Initiated Start)

Note that there is a slight difference between *Figure 83* and *Figure 84*. In the *On Demand with User Initiated Start* mode, the recording has not yet been set up. When the recording is underway, the Cisco BroadWorks Application Server sends the *a=record* attribute set to "on" in the ACK's SDP, whereas in the *On Demand* mode, the Cisco BroadWorks Application Server sends the *a=record* attribute in an INVITE's SDP.

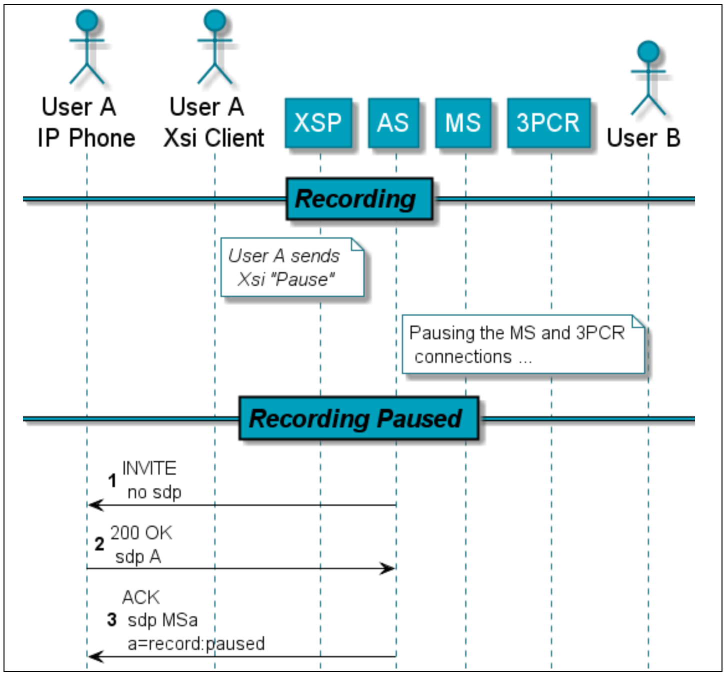

Figure 85 Xtended Services Interface Pause Recording

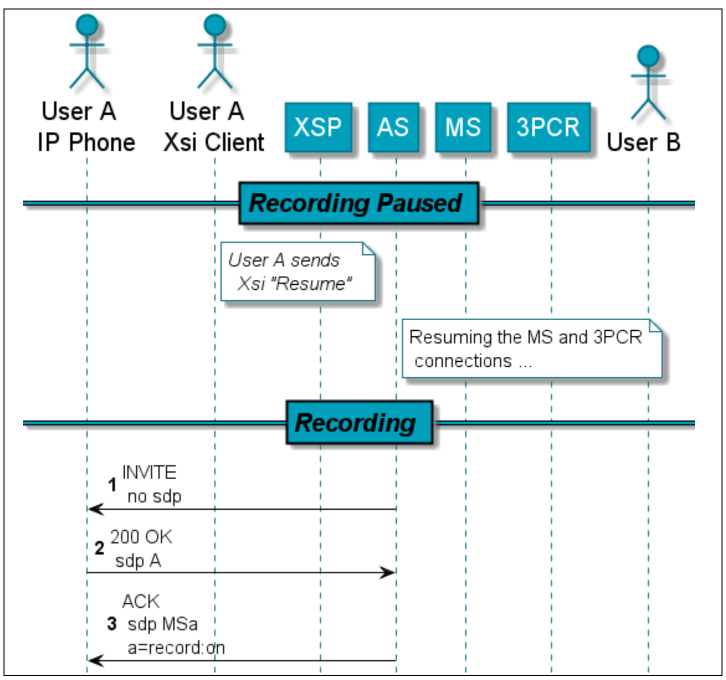

Figure 86 Xtended Services Interface Resume Recording

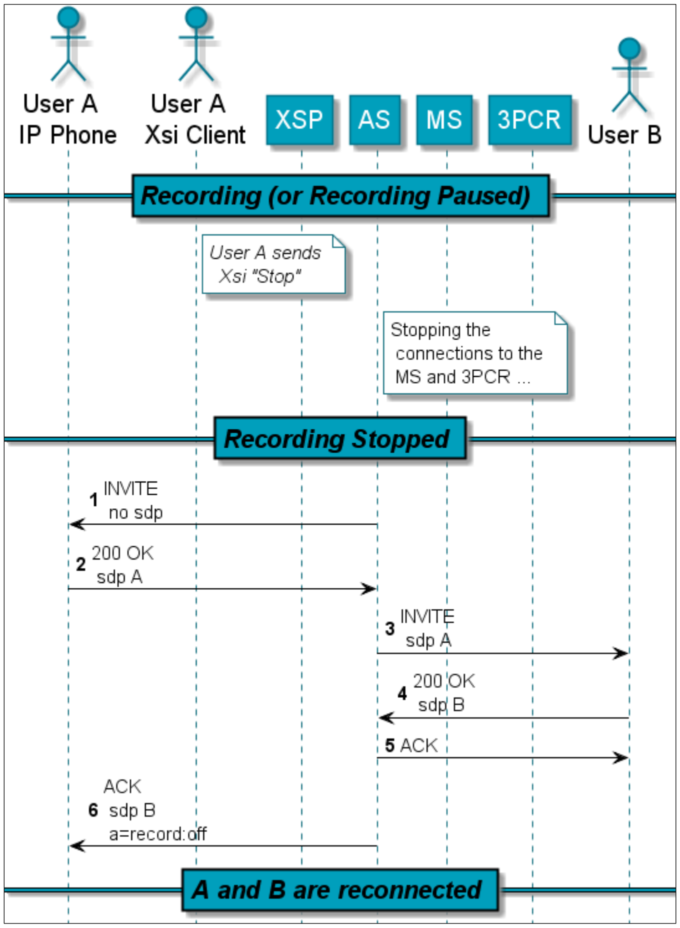

Figure 87 Xtended Services Interface Stop Recording

11 Deployment/Operational Impacts

11.1 Configuration File Impacts

There is no impact.

11.2 Installation Impacts

On installation, the new FACs are automatically present, with their default values.

FAC	Default Value
Record Call	*44
Call Recording – Stop	*45
Call Recording – Pause	*48
Call Recording – Resume	*49

11.2.1 Video Support

11.2.1.1 Application Server Installation Impacts

The system call recording parameter defined in the *Call Recording Video Support Feature Description* [14] with its added default value.

11.2.1.2 Media Server Installation Impacts

On the Media Server, a new process named *videostreammixerbe* implements the video relay function. This process is owned by root and Set Owner User ID (SUID). The file is located in */usr/local/broadworks/apps/MediaStreaming_<Release>/bin* along with other executables.

11.3 Upgrade Impacts

The four new FACs are added to the system. If, for example, the default value (*44) is already used by another FAC, the new value is left empty.

The new FACs are added to all enterprises and groups that have the feature authorized and to all users that have the feature assigned.

11.3.1.1 Start, Stop, Pause, and Resume

The *pauseResumeNotification* user service attribute is added to all users who have the feature assigned with its default value. For more information, see the *Call Recording Start Stop Pause Resume User Controls Feature Description* [12].

11.3.1.2 Video Support

11.3.1.2.1 Application Server Upgrade Impacts

The system-level call recording parameter defined with its default value in the *Call Recording Video Support Feature Description* [14].

11.3.1.2.2 Media Server Upgrade Impacts

The new centralized configuration attributes are added. The default values are as specified in the *Call Recording Video Support Feature Description* [14].

11.3.1.3 End-User Notification

The user-level Call Recording attributes defined are added with their default values to all users who have the Call Recording service assigned. See the *Call Recording End-User Notification Feature Description* [13].

NOTE: The attributes defined are only added for real users. These attributes are not applicable to virtual users.

11.3.1.4 Voice Mail Recording

The user-level call recording attribute defined is added, with its default value to all users who have the Call Recording service assigned. See the *Call Recording Voice Mail Recording Feature Description* [15].

Upon upgrade, the Call Recording service becomes a default virtual subscriber service. It only runs for virtual subscribers who have the service assigned, with the exception of Voice Messaging, where it runs if the user for whom a voice message deposit is being left has the Call Recording service.

11.3.1.5 Call Control for IP Phones

The parameter for the user-level device policies is added with its default value to all existing users in the *Call Recording Controls for IP Phones Feature Description* [16].

11.4 Rollback Impacts

The three new FACs are removed from all groups, service providers, and the system.

The Call Recording service is unassigned from all users, groups, and service providers. It is removed from all service packs, new user templates, service pack migration tasks, trunk group creation tasks, and group-assigned recording platforms, as well as system-level defined recording platforms.

11.4.1 Start, Stop, Pause, Resume

The two new call recording modes are removed.

- Users with the *Always with Pause/Resume* mode are converted to the *Always* mode.
- Users with the *On Demand with User Initiated Start* mode are converted to the *On Demand* mode.

The *pauseResumeNotification* user service attribute is removed from all users in the *Call Recording Start Stop Pause Resume User Controls Feature Description* [12].

11.4.2 Video Support

11.4.2.1 Application Server Rollback Impacts

The system-level call recording parameter defined is removed. See the *Call Recording Video Support Feature Description* [14].

11.4.2.2 Media Server Rollback Impacts

The new centralized configuration attributes added on an upgrade are removed on a rollback.

11.4.3 End-User Notification

The user-level Call Recording attributes defined are removed from all users who have the Call Recording service assigned. See the *Call Recording End User Notification Feature Description* [13].

11.4.4 Voice Mail Recording

For an existing Call Recording platform with a schema version of 3.0, the schema version is converted to the schema version 2.0.

The user-level call recording attribute is removed from all users who have the Call Recording service assigned. See the *Call Recording Voice Mail Recording Feature Description* [15].

Upon downgrade, the Call Recording service is no longer a default virtual subscriber service.

11.4.5 Call Control for IP Phones

The attribute for the user-level device policies defined is removed from all existing users. See the *Call Recording Controls for IP Phones Feature Description* [16].

12 Service Pack Information

This feature is available in the following service pack:

- Release 17.sp4

12.1 Feature Activation Impacts

12.1.1 Method of Activation

This feature is activated using the following CLI command.

```
AS_CLI/System/ActivatableFeature> activate 46941
```

12.1.2 Activatable Feature ID and Dependencies

- Activatable Feature ID: 46941.
- Activatable Feature Name: Call Recording.
- Dependencies: FR 140637 *“Enable CDR schema version R17 SP4 for Activatable Features”.*

12.1.3 Behavior Impacts upon Activation

Service provider level provisioning impacts upon activation: The Call Recording service is available to be authorized to service providers.

12.1.4 Provisioned Data Impacts on Activation

There is no impact.

12.1.5 Provisioned Data Impacts on Deactivation

Deactivation fails if the Call Recording service is authorized to any service provider.

Recording platforms are not removed.

12.1.6 OCI Command Behavior Prior to Activation

There is no impact.

13 Restrictions and Limitations

Restrictions and limitations are as follows:

- A user with the Call Recording service can only record the portion of the call in which they were present. For example, if a user barges into a call, only the portion of the call after they barged-in is recorded. Any portion of the call that occurred prior to the barge-in is not present on the user's recording.
- Calls in the process of being recorded are abruptly terminated when a failover occurs, and only a partial recording is saved to the 3PCR platform. The media connections between the Application Server and the Media Server are taken down, resulting in the call to the 3PCR server being taken down.
- To keep a recording when the user has the *On-Demand* option selected, the Cisco BroadWorks Application Server must be notified of the user's desire prior to the call ending. Notification can be done by dialing the FAC (*44) or sending an Xtended Services Interface command to the server.
- If the SessionAudit is not configured properly on the Application Server, call recordings are stopped after 20 minutes of record time. This can occur if the SessionAudit on the Application Server is disabled or if the SessionAudit timer is set to a value above 1200 seconds. The Media Server expects the cfw-dialog session to exchange a SIP dialog within a 20-minute window to keep the session active. However, if there is no dialog, the cfw-dialog session is terminated, which closes the RTP stream from the Media Server, and thus limiting the call recording to 20 minutes.
- For Video Call Support: In dual mode, the 3PCR must support the H.264 and H.263 video codecs and in single mode it must support H.264 codec. The 3PCR must support the same level of resolutions that are offered to it without downgrading them in the answer. The 3PCR must support symmetrical payload numbers, which means that it must respond with the same payload type number in the answer as in the offer.
- End-User Notification

Service	Limitation
Barge In	▪ Barged-in party is recording the barged-in call. When the barged-in party hangs up, all call recordings stop, but there is no *Recording Ended* announcement played to the remaining parties. When the barged-in party (supervisor) takes over the call, the call recording stops due to the barged-in party leaving. There is no *Recording Ended* announcement played to the remaining parties. ▪ Being barged-in party is recording the call. This is a case when a non-conference controller is recording call in conference. The call is treated as a two-party call. When the barge-in party enters or leaves the barge-in mode, the *Recording Started* announcement on behalf of the call recording on the barged party is played to the remote sides.
Call Transfer	If the transferring party has the Call Recording service in the *On Demand with User Initiated* mode and is recording calls before transfer, after the transfer is completed, there is no *Recording Ended* announcement played to the transferred party or the transfer target that is being recorded before transfer. In the blind-blind transfer (via Call Manager client or device with “Transfer + Blind”), when the user recording call is blind –blind transferred to another party, the old call recording stops and a new call recording starts when the transfer completes. The *Recording Started* announcement is replayed to the transferred party.

Service	Limitation
Conference	If a user who is not the conference controller is recording the call between the user and the conference, the call is treated as a regular two-party call. ▪ When the call recording user leaves the conference, there is no *Recording Ended* announcement played to the remaining conference parties. ▪ Multiple announcements or tones are played to the conference if multiple conference members are recording the call.
Meet-Me Conference	If a call recording user joins a Meet-Me Conference, the call is treated as a regular two-party call. When the user is recording the call in Meet-Me Conference, after the user leaves the Meet-Me Conference, there is no Recording Ended announcement played to the remaining conference parties.
Music On Hold	User A starts call recording while the remote party User B is holding the call. If Music On Hold is enabled on User B, after User B resumes the call, the Recording Started announcement is not replayed to User B. User A is listening to the Hold music and is not aware of the remotely held status.
Silent Monitoring	The supervisor has the Call Recording in the On Demand with User Initiated mode. When the supervisor starts call recording in a supervised call, the supervisor hears the Recording Started announcement, but the recording warning tone is not played.
Mid-Call Server Down	The Recording Ended announcement is not played when call recording stops due to the Media Server or the Call Recording sever going down.
Voice Portal	The Recording Started announcement is overlapped with the voice portal message. It is a limitation of call recording and reflects this feature behavior. When a user with the Call Recording in Always mode makes a call to the voice portal, the call recording starts once the user is connected to the voice portal. The voice portal message is overlapped with the Recording Started announcement.
BroadWorks Anywhere	The Recording Started announcement is overlapped with the Answer Confirmation whisper. It is a limitation of call recording and reflects this feature behavior. It only happens when the BroadWorks Anywhere location and the Call Recording is in Always mode.

- Call Control for IP phones only works with recording-aware UA devices, as defined by *draft-ietf-siprec-protocol-09* sections *7.1.2, 7.3.2, 11.1.2*, and *11.5* [19].
- Call Recording for Silent Monitoring: If the supervisor uses the *44 FAC code to start recording, they may not hear the announcement, "This call is being recorded" as this announcement plays when recording starts and the supervisor is likely still on the FAC call leg. The supervisor, however, still hears the announcement, "Your Call Recording service has successfully started".

  If the Xsi commands are used through a client to start call recording, the supervisor hears both announcements.

14 Appendix A: Troubleshooting

14.1 Unable to Activate Feature for a User

1) If the user is a group member, verify that a 3PCR platform is defined for the group.
2) Verify that the 3PCR platform is provisioned in *AS_CLI/service/CallRecording*.

14.2 No Platforms Available for a Group

1) Verify that the 3PCR platform is provisioned in *AS_CLI/service/CallRecording*.

14.3 bwCallRecordingPlatformError Alarm Raised

1) Verify IP address and port in alarm is the correct address and port of 3PCR platform.
2) Check that the 3PCR platform is active.
3) Check network connectivity between:
   – 3PCR platform and Media Server
   – Application Server and Media Server
   – Application Server and 3PCR platform

14.4 Accounting Records Show Failed

1) Check for bwCallRecordingPlatformError alarm.
2) Check that the 3PCR platform is active.
3) Verify connectivity between:
   – 3PCR platform and Media Server
   – Application Server and Media Server
   – Application Server and 3PCR platform

15 Appendix B: Interpreting XML Metadata

This section gives several examples of the XML data and the information that is included in the data. There are examples of both the dual and single modes. For details on the XML format, see section *9.6 SIP Message Body*.

15.1 Originator Call – Single Mode

This example shows a call between calling party 9726990601 and called party 9726990506. The call is being recorded on behalf of the originator of the call. This user's userid is north01 and the DN is 9726990601.

The following is the XML sent to the 3PCR at the setup of the call. The following sections show the XML data broken down into sections with the important parts highlighted. The whole XML data is included inside the *recording_metadata* element. The whole XML is shown in the following example and then broken down in the subsections that follow:


```
<recording_metadata xmlns="urn:ietf:params:xml:ns:siprec">
    <dataMode>complete</dataMode>
    <recording id="urn:uuid:f0ac6b86-ff11-49ea-a24c-ed21520ce9e0">
            <requestor>SRC</requestor>
            <type>selective</type>
    </recording>
    <group id="urn:uuid:ab8dee42-7b76-45cb-8ebf-f1e4b9339af0"
recording="urn:uuid:f0ac6b86-ff11-49ea-a24c-ed21520ce9e0">
            <initiator>sip:north01@example.broadsoft.com</initiator>
    </group>
    <session id="urn:uuid:94884940-cdbc-417b-b061-e8cd659c6a71"
group="urn:uuid:ab8dee42-7b76-45cb-8ebf-f1e4b9339af0">
            <start-time>2011-09-13T10:25:22-0500</start-time>
    </session>
    <participant id="urn:uuid:44221935-8fd8-4de9-bae6-5b7572ff341e"
session="urn:uuid:94884940-cdbc-417b-b061-e8cd659c6a71">
            <aor>sip:north01@example.broadsoft.com</aor>
            <send>
                    <id>urn:uuid:b081349c-0446-42f2-8f51-b732526639b9</id>
            </send>
    </participant>
    <participant id="urn:uuid:7880c248-2783-49c0-a273-0e8eb9eea4bb"
session="urn:uuid:94884940-cdbc-417b-b061-e8cd659c6a71">
            <aor>sip:506@example.broadsoft.com</aor>
            <send>
                    <id>urn:uuid:b081349c-0446-42f2-8f51-b732526639b9</id>
            </send>
    </participant>
    <stream id="urn:uuid:b081349c-0446-42f2-8f51-b732526639b9"
session="urn:uuid:94884940-cdbc-417b-b061-e8cd659c6a71">
            <label>1</label>
            <mode>mixed</mode>
    </stream>
    <extensiondata id="urn:uuid:7be4fa8f-2e07-4473-bf20-6dfec8e0b604"
parent="urn:uuid:94884940-cdbc-417b-b061-e8cd659c6a71">
            <broadWorksRecordingMetadata
xmlns="http://schema.broadsoft.com/broadworksCallRecording">
                    <extTrackingID>1:1</extTrackingID>
                    <serviceProviderID>TxASDev</serviceProviderID>
                    <groupID>North_as90</groupID>
                    <userID>north01@example.broadsoft.com</userID>
                    <callID>callhalf-71:0</callID>
                    <callType>
```



```
                         <origCall>

    <callingPartyNumber>sip:+19726990601@example.broadsoft.com</callingPa
rtyNumber>

    <calledPartyNumber>sip:+19726990506@example.broadsoft.com</calledPart
yNumber>

    <dialedDigits>sip:506@example.broadsoft.com</dialedDigits>
                         </origCall>
                  </callType>
                  <recordingType>on</recordingType>
           </broadWorksRecordingMetadata>
    </extensiondata>
</recording_metadata>
```

15.1.1 broadWorkRecordingMetaData

First, we will look at the Cisco BroadWorks extension data. The Cisco BroadWorks extension data is a subclass of the session component of the recording_metadata. This section of the XML tells us the following:

- By looking at the callType, we can see that this is an Orig call.
  - The number that user north01 dialed is: 506
  - The calling party number is: +19726990601
  - The called party number is: +19726990506
- The call is being recorded for user (userID field): north01@example.broadsoft.com
- The user belongs to group (groupID field): North_as90
- The user is part of ServiceProvider/Enterprise (serviceProviderID field): TxASDev
- The external tracking ID (extTrackingID field) for this call is: 1:1
- The call ID (callID field) for this call is: callhalf-71
- The recoding mode (recordingType field) is *Always* mode: on

15.1.2 Recording_Metadata

The standard SIPREC elements help to identify the participants of the call and to identify information to associate the RTP streams into the 3PCR platform. Each of the fields in the recording metadata, with the exception of the recording, points to a parent element. The recording metadata identifies all of the participants in the call, and in addition, coordinates the participant to the labeled SDP that their RTP stream is being transmitted.

In this example, the call is recording in single mode so there is only a single RTP stream for all participants. This information is coordinated through the URN UIDS in the participant element. Specifically, the send element identifies the RTP stream. The send element contains the URN UUID of the stream element that this participant's RTP is being transmitted over. Looking in the stream element reveals the label value from the SDP.

In our example, all participants are using label 1. Both participants in this call point to the stream identified by the urn:uuid:b081349c-0446-42f2-8f51-b732526639b9, shown in the following example.

```
<participant id="urn:uuid:44221935-8fd8-4de9-bae6-5b7572ff341e"
session="urn:uuid:94884940-cdbc-417b-b061-e8cd659c6a71">
           <aor>sip:north01@example.broadsoft.com</aor>
```

```
            <send>
                   <id>urn:uuid:b081349c-0446-42f2-8f51-b732526639b9</id>
            </send>
    </participant>
    <participant id="urn:uuid:7880c248-2783-49c0-a273-0e8eb9eea4bb"
session="urn:uuid:94884940-cdbc-417b-b061-e8cd659c6a71">
            <aor>sip:506@example.broadsoft.com</aor>
            <send>
                   <id>urn:uuid:b081349c-0446-42f2-8f51-b732526639b9</id>
            </send>
    </participant>
```

The stream element shows that they are using the SDP labeled with 1. In addition, the stream tells us that the SDP is mixed together since the mode of the stream is mixed.

```
<stream id="urn:uuid:b081349c-0446-42f2-8f51-b732526639b9"
session="urn:uuid:94884940-cdbc-417b-b061-e8cd659c6a71">
            <label>1</label>
            <mode>mixed</mode>
    </stream>
```

The SDP associated with this example is:

```
v=0
o=BroadWorks 20 1 IN IP4 10.16.120.22
s=-
t=0 0
m=audio 15854 RTP/AVP 0 18 9
c=IN IP4 10.16.120.22
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=sendonly
a=label:1
```

15.2 Terminator Call – Dual Mode

This example goes through a recording platform using the dual mode. This is actually the other call half of the same call as the originator example. In the example, the originator and the terminator both recorded the call. Notice that the information and the RTP streams are completely separate. The following is an example XML.

```
<recording_metadata xmlns="urn:ietf:params:xml:ns:siprec">
    <dataMode>complete</dataMode>
    <recording id="urn:uuid:12d29fde-222e-48dd-882b-585255541172">
            <requestor>SRC</requestor>
            <type>selective</type>
    </recording>
    <group id="urn:uuid:a8b9dc5f-2663-4260-8a6c-6b76431b1b17"
recording="urn:uuid:12d29fde-222e-48dd-882b-585255541172">
            <initiator>sip:south06@example.broadsoft.com</initiator>
    </group>
    <session id="urn:uuid:7901b836-5b19-4955-956b-833d9894efe0"
group="urn:uuid:a8b9dc5f-2663-4260-8a6c-6b76431b1b17">
            <start-time>2011-09-13T10:25:23-0500</start-time>
    </session>
    <participant id="urn:uuid:c052ebe0-f7c2-45b6-a3c8-7e750b99604f"
session="urn:uuid:7901b836-5b19-4955-956b-833d9894efe0">
            <aor>sip:south06@example.broadsoft.com</aor>
            <send>
```

```
                  <id>urn:uuid:b80b3c6c-1001-4429-8d03-adcdecf96ffd</id>
           </send>
    </participant>
    <participant id="urn:uuid:fc5fed71-a5ba-4c76-88db-d17eadca9b21"
session="urn:uuid:7901b836-5b19-4955-956b-833d9894efe0">
           <aor>sip:+19726990601@example.broadsoft.com</aor>
           <send>
                  <id>urn:uuid:f49d4682-1c14-45ba-9c7e-61f775e886f5</id>
           </send>
    </participant>
    <stream id="urn:uuid:b80b3c6c-1001-4429-8d03-adcdecf96ffd"
session="urn:uuid:7901b836-5b19-4955-956b-833d9894efe0">
           <label>1</label>
           <mode>separate</mode>
    </stream>
    <stream id="urn:uuid:f49d4682-1c14-45ba-9c7e-61f775e886f5"
session="urn:uuid:7901b836-5b19-4955-956b-833d9894efe0">
           <label>2</label>
           <mode>separate</mode>
    </stream>
    <extensiondata id="urn:uuid:00c99005-1d76-497b-b3c2-c1aafc72a1f5"
parent="urn:uuid:7901b836-5b19-4955-956b-833d9894efe0">
           <broadWorksRecordingMetadata
xmlns="http://schema.broadsoft.com/broadworksCallRecording">
                  <extTrackingID>1:1</extTrackingID>
                  <serviceProviderID>TxASDev</serviceProviderID>
                  <groupID>South_as90</groupID>
                  <userID>south06@example.broadsoft.com</userID>
                  <callID>callhalf-75:0</callID>
                  <callType>
                         <termCall>

    <callingPartyNumber>sip:+19726990601@example.broadsoft.com</callingPa
rtyNumber>

    <calledPartyNumber>sip:+19726990506@example.broadsoft.com</calledPart
yNumber>
                         </termCall>
                  </callType>
                  <recordingType>demand</recordingType>
           </broadWorksRecordingMetadata>
    </extensiondata>
</recording_metadata>
```

15.2.1 broadWorkRecordingMetaData

First, we will look at the Cisco BroadWorks extension data. The Cisco BroadWorks extension data is a subclass of the session component of the recording_metadata. This section of the XML tells us the following:

- By looking at the callType, we can see that this is a Term call.
  - The calling party number is: +19726990601
  - The called party number is: +19726990506
- The call is being recorded for user (userID field): south06@example.broadsoft.com.
- The user belongs to group (groupID field): South_as90.
- The user is part of ServiceProvider/Enterprise (serviceProviderID field): TxASDev.
- The external tracking ID (extTrackingID field) for this call is 1:1.

- The call ID (callID field) for this call is callhalf-71.
- The recording mode (recordingType field) is *On-Demand* mode: demand.

15.2.2 Recording_Metadata

The standard SIPREC elements help to identify the participants of the call and to identify information to associate the RTP streams into the 3PCR platform. Each of the fields in the recording metadata, with the exception of the recording, points to a parent element. The recording metadata identifies all of the participants in the call, and in addition, coordinates the participant to the labeled SDP that their RTP stream is being transmitted.

In this example, the call is recording in dual mode so there are two RTP streams for all participants. This information is coordinated through the URN UIDS in the participant element. Specifically, the send element identifies the RTP stream. The send element contains the URN UUID of the stream element that this participant's RTP is being transmitted over. Looking in the stream element reveals the label value from the SDP.

The first participant's RTP stream is identified by label 1. This is shown in the following example by coordinating the send URN uuid and the stream. The stream identifies that this is a separate stream and that it is using label 1 from the SDP at the end of this section.

```
<participant id="urn:uuid:c052ebe0-f7c2-45b6-a3c8-7e750b99604f"
session="urn:uuid:7901b836-5b19-4955-956b-833d9894efe0">
          <aor>sip:south06@example.broadsoft.com</aor>
          <send>
                 <id>urn:uuid:b80b3c6c-1001-4429-8d03-adcdecf96ffd</id>
          </send>
    </participant>

    stream id="urn:uuid:b80b3c6c-1001-4429-8d03-adcdecf96ffd"
session="urn:uuid:7901b836-5b19-4955-956b-833d9894efe0">
          <label>1</label>
          <mode>separate</mode>
    </stream>
```

The second participant's RTP stream is identified by label 2. This is shown in the following example by coordinating the send URN uuid and the stream. The stream identifies that this is a separate stream and that it is using label 2 from the SDP at the end of this section.

```
<participant id="urn:uuid:fc5fed71-a5ba-4c76-88db-d17eadca9b21"
session="urn:uuid:7901b836-5b19-4955-956b-833d9894efe0">
          <aor>sip:+19726990601@example.broadsoft.com</aor>
          <send>
                 <id>urn:uuid:f49d4682-1c14-45ba-9c7e-61f775e886f5</id>
          </send>
    </participant>

    <stream id="urn:uuid:f49d4682-1c14-45ba-9c7e-61f775e886f5"
session="urn:uuid:7901b836-5b19-4955-956b-833d9894efe0">
          <label>2</label>
          <mode>separate</mode>
    </stream>
```

The SDP associated with this example is:

```
v=0
o=BroadWorks 22 1 IN IP4 10.16.120.22
s=-
t=0 0
m=audio 15858 RTP/AVP 0 18 9
c=IN IP4 10.16.120.22
```

```
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=sendonly
a=label:1
m=audio 15862 RTP/AVP 0 18 9
c=IN IP4 10.16.120.22
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=sendonly
a=label:2
```

15.3 Conference Call

This example goes through a user creating a conference call this is bridged on the Cisco BroadWorks system. This example concentrates on the changes to the calls as the conference progresses from two individual calls and then into a three-way conference. It also shows the behavior when a party drops from the conference and the call reverts back to a two-party call. It does not go into as much detail as the previous examples but concentrates on the details specific to the conference.

15.3.1 A Calls B

This call starts when user south06 calls extension 600. The XML for this is similar to the information in the originator example. This is a call where the recording platform is running in dual mode, and the user has the On-Demand recording mode set.

Looking at the XML, you can see that the call is being recorded on behalf of user south06 and that this is an orig call with two participants:

- South06@example.broadsoft.com.
- 600@example.broadsoft.com.


```
<recording_metadata xmlns="urn:ietf:params:xml:ns:siprec">
    <dataMode>complete</dataMode>
    <recording id="urn:uuid:65ec65db-cf38-4d9f-b915-9e147ab89073">
           <requestor>SRC</requestor>
           <type>selective</type>
    </recording>
    <group id="urn:uuid:6eb89733-3e28-447a-8fcc-43cfad7ca4e4"
recording="urn:uuid:65ec65db-cf38-4d9f-b915-9e147ab89073">
           <initiator>sip:south06@example.broadsoft.com</initiator>
    </group>
    <session id="urn:uuid:f3c457fb-18ea-4a08-880f-7c36f8b1cc90"
group="urn:uuid:6eb89733-3e28-447a-8fcc-43cfad7ca4e4">
           <start-time>2011-09-13T11:11:13-0500</start-time>
    </session>
    <participant id="urn:uuid:2b286b0f-d756-420c-b2f7-c558924d90e5"
session="urn:uuid:f3c457fb-18ea-4a08-880f-7c36f8b1cc90">
           <aor>sip:south06@example.broadsoft.com</aor>
           <send>
                  <id>urn:uuid:ba0b3636-cce3-49c6-a298-e46ebcb681f9</id>
           </send>
    </participant>
    <participant id="urn:uuid:9528423e-42ae-4320-8265-46a6f86066fe"
session="urn:uuid:f3c457fb-18ea-4a08-880f-7c36f8b1cc90">
           <aor>sip:600@example.broadsoft.com</aor>
           <send>
```

```
                  <id>urn:uuid:f6e8cbe1-1443-47d0-9b2a-d6a08e3e9731</id>
           </send>
    </participant>
    <stream id="urn:uuid:ba0b3636-cce3-49c6-a298-e46ebcb681f9"
session="urn:uuid:f3c457fb-18ea-4a08-880f-7c36f8b1cc90">
           <label>1</label>
           <mode>separate</mode>
    </stream>
    <stream id="urn:uuid:f6e8cbe1-1443-47d0-9b2a-d6a08e3e9731"
session="urn:uuid:f3c457fb-18ea-4a08-880f-7c36f8b1cc90">
           <label>2</label>
           <mode>separate</mode>
    </stream>
    <extensiondata id="urn:uuid:2f8984d1-8d66-4246-954d-35b9d64ce861"
parent="urn:uuid:f3c457fb-18ea-4a08-880f-7c36f8b1cc90">
           <broadWorksRecordingMetadata
xmlns="http://schema.broadsoft.com/broadworksCallRecording">
                  <extTrackingID>4:1</extTrackingID>
                  <serviceProviderID>TxASDev</serviceProviderID>
                  <groupID>South_as90</groupID>
                  <userID>south06@example.broadsoft.com</userID>
                  <callID>callhalf-471:0</callID>
                  <callType>
                         <origCall>

    <callingPartyNumber>sip:+19726990506@example.broadsoft.com</callingPa
rtyNumber>

    <calledPartyNumber>sip:+19726990600@example.broadsoft.com</calledPart
yNumber>

    <dialedDigits>sip:600@example.broadsoft.com</dialedDigits>
                         </origCall>
                  </callType>
                  <recordingType>demand</recordingType>
           </broadWorksRecordingMetadata>
    </extensiondata>
</recording_metadata>
```

The SDP sent to the 3PCR platform is:

```
v=0
o=BroadWorks 163 1 IN IP4 10.16.120.22
s=-
t=0 0
m=audio 15968 RTP/AVP 0 18 9
c=IN IP4 10.16.120.22
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=sendonly
a=label:1
m=audio 15972 RTP/AVP 0 18 9
c=IN IP4 10.16.120.22
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=sendonly
a=label:2
```

15.3.2 A Calls C

This second call starts when user south06 call extension 601. This is a call where the recording platform is running in dual mode and the user has the On-Demand recording mode. Notice that the external tracking ID is different between this call and the previous call.

Looking at the XML, you can see that the call is being recorded on behalf of user south06 and that this is an orig call with two participants:

- South06@example.broadsoft.com
- 601@example.broadsoft.com

This example shows the full multipart mime body that the 3PCR platform will actually receive.


```
--UniqueBroadWorksBoundary
Content-Type:application/rs-metadata+xml
Content-Disposition:recording-session
Content-Length:2178

<?xml version="1.0" encoding="UTF-8"?>
<recording_metadata xmlns="urn:ietf:params:xml:ns:siprec">
    <dataMode>complete</dataMode>
    <recording id="urn:uuid:2686c78c-ce2f-4a67-bf51-7e6759409112">
            <requestor>SRC</requestor>
            <type>selective</type>
    </recording>
    <group id="urn:uuid:b6197c8b-8dd4-4acd-8fbe-ee7f0b194173"
recording="urn:uuid:2686c78c-ce2f-4a67-bf51-7e6759409112">
            <initiator>sip:south06@orton.rtx.broadsoft.com</initiator>
    </group>
    <session id="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7"
group="urn:uuid:b6197c8b-8dd4-4acd-8fbe-ee7f0b194173">
            <start-time>2011-09-13T11:11:25-0500</start-time>
    </session>
    <participant id="urn:uuid:11697232-62ec-45d9-b993-af635c6c0d9c"
session="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7">
            <aor>sip:south06@orton.rtx.broadsoft.com</aor>
            <send>
                    <id>urn:uuid:e961a6e8-db2f-4626-b9dd-fc66c754931b</id>
            </send>
    </participant>
    <participant id="urn:uuid:9f04d67b-2b61-487f-8d03-9694c9c331cd"
session="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7">
            <aor>sip:601@orton.rtx.broadsoft.com</aor>
            <send>
                    <id>urn:uuid:0c4f1abd-c92e-4b3b-b344-91b25ecb7137</id>
            </send>
    </participant>
    <stream id="urn:uuid:e961a6e8-db2f-4626-b9dd-fc66c754931b"
session="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7">
            <label>1</label>
            <mode>separate</mode>
    </stream>
    <stream id="urn:uuid:0c4f1abd-c92e-4b3b-b344-91b25ecb7137"
session="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7">
            <label>2</label>
            <mode>separate</mode>
    </stream>
    <extensiondata id="urn:uuid:9b347762-eeff-492f-9f4d-cbaa98d1651b"
parent="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7">
```

```
            <broadWorksRecordingMetadata
xmlns="http://schema.broadsoft.com/broadworksCallRecording">
                    <extTrackingID>5:1</extTrackingID>
                    <serviceProviderID>TxASDev</serviceProviderID>
                    <groupID>South_as90</groupID>
                    <userID>south06@orton.rtx.broadsoft.com</userID>
                    <callID>callhalf-471:1</callID>
                    <callType>
                           <origCall>

    <callingPartyNumber>sip:+19726990506@orton.rtx.broadsoft.com</calling
PartyNumber>

    <calledPartyNumber>sip:+19726990601@orton.rtx.broadsoft.com</calledPa
rtyNumber>

    <dialedDigits>sip:601@orton.rtx.broadsoft.com</dialedDigits>
                           </origCall>
                    </callType>
                    <recordingType>demand</recordingType>
            </broadWorksRecordingMetadata>
    </extensiondata>
</recording_metadata>
--UniqueBroadWorksBoundary
Content-Type:application/sdp
Content-Length:367

v=0
o=BroadWorks 173 1 IN IP4 10.16.120.22
s=-
t=0 0
m=audio 15988 RTP/AVP 0 18 9
c=IN IP4 10.16.120.22
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=sendonly
a=label:1
m=audio 15992 RTP/AVP 0 18 9
c=IN IP4 10.16.120.22
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=sendonly
a=label:2

--UniqueBroadWorksBoundary--
```


15.3.3 A Initiates Conference

When user south06 initiates the conference, the information for each call at the call recording platform is updated. There are still only two calls, but looking at the participant fields, they show three users for both calls. In addition, the SDP streams change while the SDP for user the call is being recorded on behalf is still using label one. Now both of the other participants are using label 2.

The following example shows the updated XML for A calling B. Notice that the external tracking ID is the same and that there are now three participants in the call. Also notice that the participants north01 and 600 are now using label 2 of the SDP.


```
--UniqueBroadWorksBoundary
Content-Type:application/rs-metadata+xml
Content-Disposition:recording-session
Content-Length:2421

<?xml version="1.0" encoding="UTF-8"?>
<recording_metadata xmlns="urn:ietf:params:xml:ns:siprec">
    <dataMode>complete</dataMode>
    <recording id="urn:uuid:65ec65db-cf38-4d9f-b915-9e147ab89073">
            <requestor>SRC</requestor>
            <type>selective</type>
    </recording>
    <group id="urn:uuid:6eb89733-3e28-447a-8fcc-43cfad7ca4e4"
recording="urn:uuid:65ec65db-cf38-4d9f-b915-9e147ab89073">
            <initiator>sip:south06@orton.rtx.broadsoft.com</initiator>
    </group>
    <session id="urn:uuid:f3c457fb-18ea-4a08-880f-7c36f8b1cc90"
group="urn:uuid:6eb89733-3e28-447a-8fcc-43cfad7ca4e4">
            <start-time>2011-09-13T11:11:13-0500</start-time>
    </session>
    <participant id="urn:uuid:2b286b0f-d756-420c-b2f7-c558924d90e5"
session="urn:uuid:f3c457fb-18ea-4a08-880f-7c36f8b1cc90">
            <aor>sip:south06@orton.rtx.broadsoft.com</aor>
            <send>
                    <id>urn:uuid:ba0b3636-cce3-49c6-a298-e46ebcb681f9</id>
            </send>
    </participant>
    <participant id="urn:uuid:9528423e-42ae-4320-8265-46a6f86066fe"
session="urn:uuid:f3c457fb-18ea-4a08-880f-7c36f8b1cc90">
            <aor>sip:600@orton.rtx.broadsoft.com</aor>
            <send>
                    <id>urn:uuid:f6e8cbe1-1443-47d0-9b2a-d6a08e3e9731</id>
            </send>
    </participant>
    <participant id="urn:uuid:72c85085-48fc-4537-a062-618b4a32885a"
session="urn:uuid:f3c457fb-18ea-4a08-880f-7c36f8b1cc90">
            <aor>north01@orton.rtx.broadsoft.com</aor>
            <send>
                    <id>urn:uuid:f6e8cbe1-1443-47d0-9b2a-d6a08e3e9731</id>
            </send>
    </participant>
    <stream id="urn:uuid:ba0b3636-cce3-49c6-a298-e46ebcb681f9"
session="urn:uuid:f3c457fb-18ea-4a08-880f-7c36f8b1cc90">
            <label>1</label>
            <mode>separate</mode>
    </stream>
    <stream id="urn:uuid:f6e8cbe1-1443-47d0-9b2a-d6a08e3e9731"
session="urn:uuid:f3c457fb-18ea-4a08-880f-7c36f8b1cc90">
            <label>2</label>
            <mode>separate</mode>
    </stream>
    <extensiondata id="urn:uuid:2f8984d1-8d66-4246-954d-35b9d64ce861"
parent="urn:uuid:f3c457fb-18ea-4a08-880f-7c36f8b1cc90">
            <broadWorksRecordingMetadata
xmlns="http://schema.broadsoft.com/broadworksCallRecording">
                    <extTrackingID>4:1</extTrackingID>
                    <serviceProviderID>TxASDev</serviceProviderID>
                    <groupID>South_as90</groupID>
```

```
                    <userID>south06@orton.rtx.broadsoft.com</userID>
                    <callID>callhalf-471:0</callID>
                    <callType>
                           <origCall>

    <callingPartyNumber>sip:+19726990506@orton.rtx.broadsoft.com</calling
PartyNumber>

    <calledPartyNumber>sip:+19726990600@orton.rtx.broadsoft.com</calledPa
rtyNumber>

    <dialedDigits>sip:600@orton.rtx.broadsoft.com</dialedDigits>
                           </origCall>
                    </callType>
                    <recordingType>demand</recordingType>
           </broadWorksRecordingMetadata>
    </extensiondata>
</recording_metadata>
--UniqueBroadWorksBoundary
Content-Type:application/sdp
Content-Length:367

v=0
o=BroadWorks 190 1 IN IP4 10.16.120.22
s=-
t=0 0
m=audio 16030 RTP/AVP 0 18 9
c=IN IP4 10.16.120.22
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=sendonly
a=label:1
m=audio 16034 RTP/AVP 0 18 9
c=IN IP4 10.16.120.22
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=sendonly
a=label:2

--UniqueBroadWorksBoundary--
```

This example shows the XML for A calls C. Notice that the external tracking ID is the same and that there are now three participants in the call. Also notice that the participants north00 and 601 are now using label 2 of the SDP.

```
--UniqueBroadWorksBoundary
Content-Type:application/rs-metadata+xml
Content-Disposition:recording-session
Content-Length:2421

<?xml version="1.0" encoding="UTF-8"?>
<recording_metadata xmlns="urn:ietf:params:xml:ns:siprec">
    <dataMode>complete</dataMode>
    <recording id="urn:uuid:2686c78c-ce2f-4a67-bf51-7e6759409112">
           <requestor>SRC</requestor>
           <type>selective</type>
    </recording>
```

```
    <group id="urn:uuid:b6197c8b-8dd4-4acd-8fbe-ee7f0b194173"
recording="urn:uuid:2686c78c-ce2f-4a67-bf51-7e6759409112">
          <initiator>sip:south06@orton.rtx.broadsoft.com</initiator>
    </group>
    <session id="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7"
group="urn:uuid:b6197c8b-8dd4-4acd-8fbe-ee7f0b194173">
          <start-time>2011-09-13T11:11:25-0500</start-time>
    </session>
    <participant id="urn:uuid:11697232-62ec-45d9-b993-af635c6c0d9c"
session="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7">
          <aor>sip:south06@orton.rtx.broadsoft.com</aor>
          <send>
                <id>urn:uuid:e961a6e8-db2f-4626-b9dd-fc66c754931b</id>
          </send>
    </participant>
    <participant id="urn:uuid:9f04d67b-2b61-487f-8d03-9694c9c331cd"
session="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7">
          <aor>sip:601@orton.rtx.broadsoft.com</aor>
          <send>
                <id>urn:uuid:0c4f1abd-c92e-4b3b-b344-91b25ecb7137</id>
          </send>
    </participant>
    <participant id="urn:uuid:3eb2318d-dff6-4c9c-ad74-3108c508b606"
session="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7">
          <aor>north00@orton.rtx.broadsoft.com</aor>
          <send>
                <id>urn:uuid:0c4f1abd-c92e-4b3b-b344-91b25ecb7137</id>
          </send>
    </participant>
    <stream id="urn:uuid:e961a6e8-db2f-4626-b9dd-fc66c754931b"
session="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7">
          <label>1</label>
          <mode>separate</mode>
    </stream>
    <stream id="urn:uuid:0c4f1abd-c92e-4b3b-b344-91b25ecb7137"
session="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7">
          <label>2</label>
          <mode>separate</mode>
    </stream>
    <extensiondata id="urn:uuid:9b347762-eeff-492f-9f4d-cbaa98d1651b"
parent="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7">
          <broadWorksRecordingMetadata
xmlns="http://schema.broadsoft.com/broadworksCallRecording">
                <extTrackingID>5:1</extTrackingID>
                <serviceProviderID>TxASDev</serviceProviderID>
                <groupID>South_as90</groupID>
                <userID>south06@orton.rtx.broadsoft.com</userID>
                <callID>callhalf-471:1</callID>
                <callType>
                      <origCall>

    <callingPartyNumber>sip:+19726990506@orton.rtx.broadsoft.com</calling
PartyNumber>

    <calledPartyNumber>sip:+19726990601@orton.rtx.broadsoft.com</calledPa
rtyNumber>

    <dialedDigits>sip:601@orton.rtx.broadsoft.com</dialedDigits>
                      </origCall>
                </callType>
                <recordingType>demand</recordingType>
          </broadWorksRecordingMetadata>
```

```
    </extensiondata>
</recording_metadata>
--UniqueBroadWorksBoundary
Content-Type:application/sdp
Content-Length:367

v=0
o=BroadWorks 188 1 IN IP4 10.16.120.22
s=-
t=0 0
m=audio 16022 RTP/AVP 0 18 9
c=IN IP4 10.16.120.22
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=sendonly
a=label:1
m=audio 16026 RTP/AVP 0 18 9
c=IN IP4 10.16.120.22
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=sendonly
a=label:2

--UniqueBroadWorksBoundary--
```

15.3.4 B Releases

In this example, the last change to the call is that party B drops from the conference call. This causes the conference to be reconfigured back to a two-party call. The XML data changes, for A calls C, to reflect this change in the call. The A calls B call is released from the call recording platform. Notice that in the XML data, while there are still three participants in the call, only two of the participants have a send element. The participant that dropped from the conference no longer has an active send field.

```
--UniqueBroadWorksBoundary
Content-Type:application/rs-metadata+xml
Content-Disposition:recording-session
Content-Length:2421

<?xml version="1.0" encoding="UTF-8"?>
<recording_metadata xmlns="urn:ietf:params:xml:ns:siprec">
    <dataMode>complete</dataMode>
    <recording id="urn:uuid:2686c78c-ce2f-4a67-bf51-7e6759409112">
            <requestor>SRC</requestor>
            <type>selective</type>
    </recording>
    <group id="urn:uuid:b6197c8b-8dd4-4acd-8fbe-ee7f0b194173"
recording="urn:uuid:2686c78c-ce2f-4a67-bf51-7e6759409112">
            <initiator>sip:south06@orton.rtx.broadsoft.com</initiator>
    </group>
    <session id="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7"
group="urn:uuid:b6197c8b-8dd4-4acd-8fbe-ee7f0b194173">
            <start-time>2011-09-13T11:11:25-0500</start-time>
    </session>
    <participant id="urn:uuid:11697232-62ec-45d9-b993-af635c6c0d9c"
session="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7">
            <aor>sip:south06@orton.rtx.broadsoft.com</aor>
```

```
          <send>
                 <id>urn:uuid:e961a6e8-db2f-4626-b9dd-fc66c754931b</id>
          </send>
   </participant>
   <participant id="urn:uuid:9f04d67b-2b61-487f-8d03-9694c9c331cd"
session="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7">
          <aor>sip:601@orton.rtx.broadsoft.com</aor>
          <send>
                 <id>urn:uuid:0c4f1abd-c92e-4b3b-b344-91b25ecb7137</id>
          </send>
   </participant>
   <participant id="urn:uuid:3eb2318d-dff6-4c9c-ad74-3108c508b606"
session="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7">
          <aor>north00@orton.rtx.broadsoft.com</aor>
   </participant>
   <stream id="urn:uuid:e961a6e8-db2f-4626-b9dd-fc66c754931b"
session="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7">
          <label>1</label>
          <mode>separate</mode>
   </stream>
   <stream id="urn:uuid:0c4f1abd-c92e-4b3b-b344-91b25ecb7137"
session="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7">
          <label>2</label>
          <mode>separate</mode>
   </stream>
   <extensiondata id="urn:uuid:9b347762-eeff-492f-9f4d-cbaa98d1651b"
parent="urn:uuid:cbe4db33-9a47-4f8c-b74b-985c09cb1be7">
          <broadWorksRecordingMetadata
xmlns="http://schema.broadsoft.com/broadworksCallRecording">
                 <extTrackingID>5:1</extTrackingID>
                 <serviceProviderID>TxASDev</serviceProviderID>
                 <groupID>South_as90</groupID>
                 <userID>south06@orton.rtx.broadsoft.com</userID>
                 <callID>callhalf-471:1</callID>
                 <callType>
                        <origCall>

   <callingPartyNumber>sip:+19726990506@orton.rtx.broadsoft.com</calling
PartyNumber>

   <calledPartyNumber>sip:+19726990601@orton.rtx.broadsoft.com</calledPa
rtyNumber>

   <dialedDigits>sip:601@orton.rtx.broadsoft.com</dialedDigits>
                        </origCall>
                 </callType>
                 <recordingType>demand</recordingType>
          </broadWorksRecordingMetadata>
   </extensiondata>
</recording_metadata>
--UniqueBroadWorksBoundary
Content-Type:application/sdp
Content-Length:367

v=0
o=BroadWorks 192 1 IN IP4 10.16.120.22
s=-
t=0 0
m=audio 16038 RTP/AVP 0 18 9
c=IN IP4 10.16.120.22
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
```

```
a=rtpmap:9 G722/8000
a=ptime:20
a=sendonly
a=label:1
m=audio 16042 RTP/AVP 0 18 9
c=IN IP4 10.16.120.22
a=rtpmap:0 PCMU/8000
a=rtpmap:18 G729/8000
a=rtpmap:9 G722/8000
a=ptime:20
a=sendonly
a=label:2


--UniqueBroadWorksBoundary-
```

15.4 Visual Security Classification Call

Visual Security Classification impacts the *broadworks-recording-metadata* definition of the 3PCR platform interface by adding the *callClassification* element with the call classification security value.

The *broadworks-recording-metadata* is used in the message body in two places.

- The first usage is in the call setup to the 3PCR platform. In this instance, the data contains information from the Cisco BroadWorks Application Server, which identifies the call. This information should be used by the 3PCR platform to index and store the call recording.
- The second usage of this data is in the re-INVITE or the UPDATE message when the metadata is being updated due to a change in the call being recorded.

For interoperability, this new element is only included in the metadata sent to the 3PCR platforms that support *broadworks-recording-metadata* schema version 3.0 or later. The administrator can indicate the schema version against each 3PCR platform configuration, as described in section *6.3.5 Add Call Recording Platform(s)*.

The following schema definition shows the new element highlighted in **bold** font. For the previous schema definition (version 2.0), see the *Call Recording – Add Call Center Metadata Feature Description* [23].

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
targetNamespace="http://schema.broadsoft.com/broadworksCallRecording"
  xmlns:tns="http://schema.broadsoft.com/broadworksCallRecording"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified"
  version="3.0">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
  schemaLocation="http://www.w3.org/2001/xml.xsd"/>
  <xs:element name="BroadWorks-Recording-metadata"
  type="tns:BroadWorks-recording-metadata"/>
  <xs:complexType name="broadWorksRecordingMetadata">
    <xs:sequence>
      <xs:element name="extTrackingID" type="xs:string"
        minOccurs="1"/>
      <xs:element name="serviceProviderID" type="xs:string"
        minOccurs="0"/>
      <xs:element name="groupID" type="xs:string" minOccurs="0"/>
```

```
      <xs:element name="userID" type="xs:string" minOccurs="0"/>
      <xs:element name="callID" type="xs:string" minOccurs="0"/>
      <xs:element name="callType" type="tns:callTypeInfo" minOccurs="0"/>
      <xs:element name="recordingType" type="tns:recordingType"
        minOccurs="0"/>
      <xs:element name="redirectedCall" type="tns:redirectedCall"
        minOccurs="0"/>
      <xs:element name="acd" type="tns:acdDetails" minOccurs="0"/>
      <xs:element name="callClassification" type="xs:string"
minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="callTypeInfo">
    <xs:choice>
      <xs:element name="origCall" type="tns:origCallDetails"/>
      <xs:element name="termCall" type="tns:termCallDetails"/>
    </xs:choice>
  </xs:complexType>
  <xs:complexType name="origCallDetails">
    <xs:sequence>
      <xs:element name="callingPartyNumber" type="xs:string" />
      <xs:element name="calledPartyNumber" type="xs:string"/>
      <xs:element name="dialedDigits" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="termCallDetails">
    <xs:sequence>
      <xs:element name="callingPartyNumber" type="xs:string"/>
      <xs:element name="calledPartyNumber" type="xs:string"/>
      <xs:element name="redirectInfo" type="tns:redirectInfo"
        minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="redirectInfo">
    <xs:sequence>
      <xs:element name="lastRedirectNumber" type="xs:string"
        minOccurs="0"/>
      <xs:element name="origCalledNumber" type="xs:string"
        minOccurs="0"/>
      <xs:element name="numOfRedirections" type="xs:integer"
        minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="redirectedCall">
    <xs:sequence>
      <xs:element name="newExtTrackingID" type="xs:string"
        minOccurs="0"/>
      <xs:element name="redirectedFromPartyNumber" type="xs:string"/>
      <xs:element name="redirectedToPartyNumber" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="recordingType">
    <xs:restriction base="xs:string">
      <xs:pattern value="on|demand"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="acdDetails">
    <xs:sequence>
      <xs:element name="acdUserId" type="xs:string" />
      <xs:element name="acdName" type="xs:string"/>
      <xs:element name="acdNumber" type="tns:acdNumberDetails"/>
    </xs:sequence>
```

```
  </xs:complexType>
  <xs:complexType name="acdNumberDetails">
    <xs:simpleContent>
      <xs:extension base="NonEmptyURI">
        <xs:attribute name="countryCode" type="xs:string"
use="optional"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:simpleType name="NonEmptyURI">
    <xs:restriction base="xs:anyURI">
      <xs:minLength value="1"/>
    </xs:restriction>
  </xs:simpleType>
</xs:schema>
```

References

[1] Portman, L., Lum, H., Johnston, A., Hutton, A., "The SIP-based Media Recording Protocol (SIPREC)", draft-portman-siprec-protocol-03, Internet Engineering Task Force, March 1, 2011. Available from http://www.ietf.org.

[2] Ravindranath, R., Ravindran, P., Kyzivat, P., "Session Initiation Protocol (SIP) Recording Metadata Format", draft-ram-siprec-metadata-format-01, Internet Engineering Task Force, March 8, 2011. Available from http://www.ietf.org.

[3] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J, Sparks R., Handley, M., and Schooler, E., "SIP: Session Initiation Protocol", RFC 3261, Internet Engineering Task Force, June 2002. Available from http://www.ietf.org.

[4] Levin, O., Camarillo, G., "The Session Description Protocol (SDP) Label Attribute", RFC 4574, August 2006. Available from http://www.ietf.org.

[5] Holmberg, C., Burger, E., Kaplan, H., "Session Initiation Protocol (SIP) INFO Method and Package Framework", RFC 6086, January 2011. Available from http://www.ietf.org.

[6] Schulzrinne, H., Petrack, S., "RTP Payload for DTMF Digits, TelephoneTones and Telephony Signals", RFC 2833, May 2000. Available from http://www.ietf.org.

[7] Schulzrinne, H., Taylor, T., "RTP Payload for DTMF Digits, TelephoneTones and Telephony Signals", RFC 4733, December 2006. Available from http://www.ietf.org.

[8] Schulzrinne, H., Taylor, T., "Definition of Events for Modem Fax, and Text Telephony Signals", RFC 4734, December 2006. Available from http://www.ietf.org.

[9] Cisco Systems, Inc. 2016. *BroadWorks Media Server EMS Configuration Management Guide, Release 22.0.* Available from Cisco at xchange.broadsoft.com.

[10] Cisco Systems, Inc. 2019. *Cisco BroadWorks Call Center Solution Guide, Release 23.0*. Available from Cisco at xchange.broadsoft.com.

[11] Cisco Systems, Inc. 2012. *Network CTI Integration Feature Description, Release 17.0*. Available from BroadSoft at xchange.broadsoft.com.

[12] BroadSoft, Inc. 2013. *Call Recording Start Stop Pause Resume User Control Feature Description, Release 20.0*. Available from Cisco at xchange.broadsoft.com.

[13] Cisco Systems, Inc. 2013. *Call Recording End-User Notification of Recording Feature Description, Release 20.0*. Available from Cisco at xchange.broadsoft.com.

[14] Cisco Systems, Inc. 2013. *Call Recording Video Support Feature Description, Release 20.0*. Available from Cisco at xchange.broadsoft.com.

[15] Cisco Systems, Inc. 2013. *Call Recording Voice Mail Recording Feature Description, Release 20.0*. Available from Cisco at xchange.broadsoft.com.

[16] Cisco Systems, Inc. 2013. *Call Recording Controls for IP Phones Feature Description, Release 20.0*. Available from Cisco at xchange.broadsoft.com.

[17] Cisco Systems, Inc. 2013. *Call Recording Visual Security Classification for Active Call Feature Description, Release 20.0*. Available from Cisco at xchange.broadsoft.com.

[18] Cisco Systems, Inc. 2013. *Media Server Performance Optimizations for Call Recording Port Usage Feature Description, Release 20.0*. Available from Cisco at xchange.broadsoft.com.

[19] Portman, L., Lum, H., Eckel, C., Johnston, A., and Hutton, A., “Session Recording Protocol”, draft-ietf-siprec-protocol-09, Internet Engineering Task Force, December 2012. Available from http://www.ietf.org/.

[20] Levin, O., Even, R., Hagendorf, P., “XML Schema for Media Control”, *RFC 5168*, March, 2008. Available from http://www.ietf.org.

[21] Cisco Systems, Inc. 2013. *Video Conferencing Support Feature Description*, *Release 20.0*. Available from Cisco at xchange.broadsoft.com.

[22] Cisco Systems, Inc. 2013. *Media Server – Performance Optimizations for Call Recording Port Usage Feature Description*, *Release 20.0*. Available from Cisco at xchange.broadsoft.com.

[23] Cisco Systems, Inc. 2012. *Call Recording – Add Call Center Metadata Feature Description, Release 19.0*. Available from Cisco at xchange.broadsoft.com.

[24] Cisco Systems, Inc. 2014. *Media Server Redundancy for Call Recording and N-Way Conferencing Feature Description*, *Release 21.0.* Available from Cisco at xchange.broadsoft.com.

[25] Rosenberg, J., “Session Initiation Protocol (SIP) UPDATE Method”, RFC 3311, Internet Engineering Task Force, September 2002. Available from http://www.ietf.org.

[26] Gulbrandsen, A., Vixie, P., Esibov, L., “A DNS RR for specifying the location of services (DNS SRV)” , RFC 2782, Internet Engineering Task Force, June 2002. Available from www.ietf.org.

[27] Cisco Systems, Inc. 2016. *SRV Support for Call Recording Platforms Feature Description, Release 22.0*. Available from Cisco at xchange.broadsoft.com.

Acronyms and Abbreviations

3PCR	Third Party Call Recording
ABNF	Augmented Backus-Naur Form
AS	Application Server
CDR	Call Detail Record
CFA	Call Forwarding Always
CFB	Call Forwarding Busy
CFW	Control Channel Framework
CLI	Command Line Interface
CTI	Computer Telephony Integration
CR	Call Recording
DN	Directory Number
DNIS	Dialed Number Identification Service
DNS	Domain Name System
DTMF	Dual-tone Multi-frequency
EMS	Element Management System
FAC	Feature Access Code
FQDN	Fully Qualified Domain Name
GMT	Greenwich Mean Time
IANA	Internet Assigned Numbers Authority
IETF	Internet Engineering Task Force
IVR	Interactive Voice Response
PBX	Private Branch Exchange
RFC	Request for Comments
RTP	Real-Time Transport Protocol
SDP	Session Description Protocol
SIP	Session Initiation Protocol
SIPREC	SIP Recording
SRV	Service Locator
UA	User Agent
UUID	Universally Unique Identifier
XML	eXtensible Markup Language
XSD	XML Schema Definition
Xsi	Xtended Services Interface
Xsp	Xtended Services Platform