

TR-1069

SIP によるリモート通話録音
に関する技術報告書

Technical Report on SIP-Based Media Recording

第1版

2018年1月30日制定

一般社団法人
情報通信技術委員会

THE TELECOMMUNICATION TECHNOLOGY COMMITTEE

本書は、一般社団法人情報通信技術委員会が著作権を保有しています。
内容の一部又は全部を一般社団法人情報通信技術委員会の許諾を得ることなく複製、転載、
改変、転用及びネットワーク上での送信、配布を行うことを禁止します。

目次

<参考>.....	5
1. 概説.....	6
1.1 本報告書の適用範囲.....	6
1.1.1 基本接続形態.....	6
1.2 本報告書の目的と規定.....	6
1.3 本報告書の内容.....	7
1.3.1 本報告書の策定の背景と位置づけについて.....	7
1.3.2 本報告書の位置づけ.....	7
1.4 用語.....	7
2. セッション録音アーキテクチャ (RFC7245 第3章).....	9
2.1 SRC の役割.....	9
2.1.1 B2BUA を SRC にする場合.....	9
2.1.2 エンドポイント を SRC にする場合.....	9
2.1.3 SIP プロキシは SRC には不適.....	10
2.2 RS の確立.....	10
2.2.1 SRC による録音の開始.....	10
2.2.2 SRS による録音の開始.....	10
2.2.3 RS の一時停止/再開.....	11
2.2.4 メディアストリームのミキシング.....	11
2.2.5 メディアのトランスコード.....	11
2.2.6 損失のない録音.....	11
3. 録音シーケンス (RFC7866 第5章).....	12
3.1 記録メディアの配信.....	12
3.2 録音メタデータの配信.....	14
3.3 録音表示の受信と録音の提供.....	16
4. SIP ハンドリング (RFC7866 第6章).....	17
4.1 SRC における手順.....	17
4.1.1 RS の開始.....	17
4.1.2 録音の指示と設定のための SIP 拡張.....	17
4.2 SRS における手順.....	17
4.3 RECORDING-AWARE UA のための手順.....	17
5. SDP ハンドリング (RFC7866 第7章).....	18
5.1 SRC における手順.....	18
5.1.1 RS における SDP ハンドリング.....	18
5.1.2 CS における録音識別.....	19
5.1.3 CS での録音優先.....	19
5.2 SRS における手順.....	19
5.3 RECORDING-AWARE UA のための手順.....	21
5.3.1 録音識別.....	21
5.3.2 録音選択.....	21

6. RTP ハンドリング (RFC7866 第 8 章).....	23
6.1 RTP メカニズム.....	23
6.1.1 RTCP.....	23
6.1.2 RTP プロフィール.....	23
6.1.3 SSRC.....	23
6.1.4 CSRC	24
6.1.5 SDES.....	24
6.1.6 Keepalive	24
6.1.7 RTCP フィードバックメッセージ	24
6.1.8 送信と受信に関するシンメトリック RTP/RTCP	25
6.2 役割.....	25
6.2.1 RTP トランスレータとして機能する SRC	26
6.2.2 RTP ミキサーとして機能する SRC	27
6.2.3 RTP エンドポイントとして機能する SRC	27
6.3 SRC による RTP セッション用法	27
6.3.1 複数の m 行を使用する SRC	28
6.3.2 ミキシングを使用する SRC	28
6.4 SRS による RTP セッション用法.....	29
7. メタデータ (RFC7866 第 9 章).....	30
7.1 SRC の手順	30
7.2 SRS の手順.....	31

<参考>

1. 国際勧告等の関連

本技術レポートは、The Internet Society において制定された「Use Cases and Requirements for SIP-Based Media Recording (SIPREC) (RFC6342)」「An Architecture for Media Recording Using the Session Initiation Protocol (RFC7245)」「Session Initiation Protocol (SIP) Recording Metadata (RFC7865)」「Session Recording Protocol (RFC7866)」に準拠している。

2. 上記国際勧告等に対する追加項目等

なし

3. 改版の履歴

版数	制定日	改版内容
第1版	2018年1月30日	制定

4. 工業所有権

本標準に関わる「工業所有権等の実施の権利に係る確認書」の提出状況は、TTC ホームページでご覧になれます。

5. 技術レポート作成部門

第1版 : 企業ネットワーク専門委員会

1. 概説

1.1 本報告書の適用範囲

本報告書は、JJ-22.01 [2] に規定されるフレームワーク標準の網接続アーキテクチャにおいて、セッション録音サーバ (Session Recording Server (SRS)) および SIP (Session Initiation Protocol) 端末などのセッション録音クライアント (Session Recording Client (SRC)) 間 (インタフェース E) の SIP を用いた録音セッション (Recording Session (RS)) の確立における推奨仕様を規定するものである。

また、インタフェース B を経由して接続される端末が、本報告書の範囲を超えた能力を保持することを妨げるものではない。但し、その場合においても本報告書に準拠する端末との接続性について考慮することが望ましい。

1.1.1 基本接続形態

本報告書は、図 1.1 で示す企業 SIP 網相互接続モデルに規定されるインタフェース E に適用可能な管理された企業 SIP 網との接続インタフェースの条件を示す。本インタフェースの規定を遵守できるインタフェースを有する企業 SIP 網に関して、

本報告書では“管理された企業 SIP 網”と呼ぶ。以下企業 SIP 網と表記する場合は、“管理された企業 SIP 網”であることを前提とする。

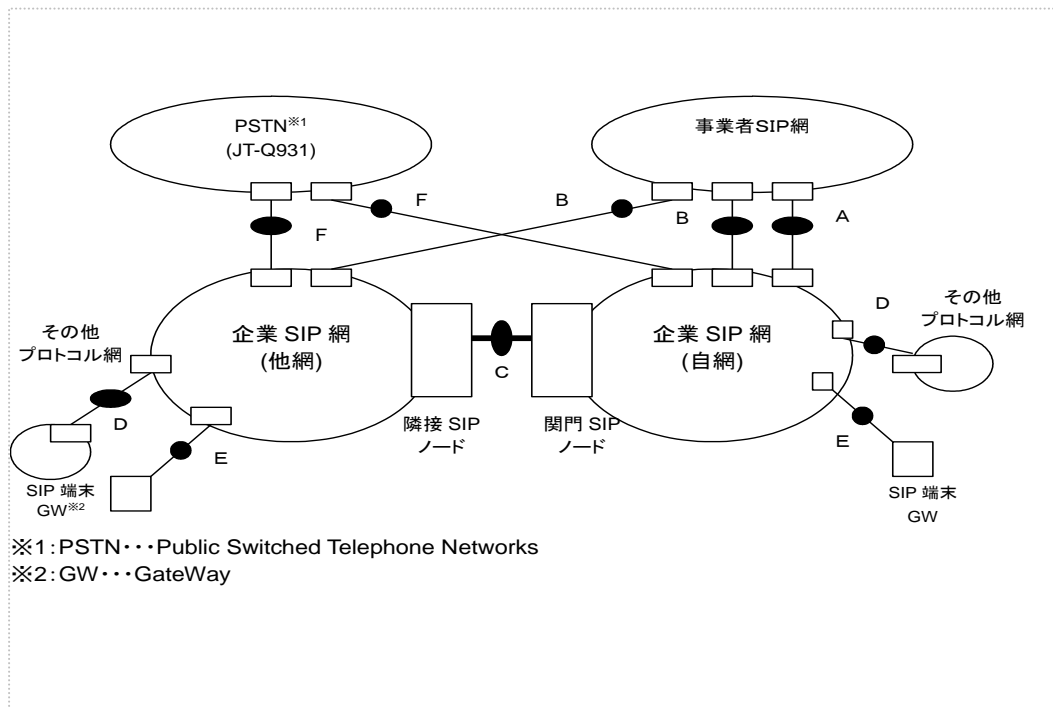


図 1.1: 企業 SIP 網相互接続モデル

1.2 本報告書の目的と規定

本報告書では、SRS、SRC の実装に際して、

- 接続条件に関わる規定の解釈を一意とすることで、実装可能な標準とする。
- SIP 網を介した SRS および SRC の接続において、共通的に適用することが可能な標準とする。
- 本規定の範囲を超えるまたは、厳密に本規定を遵守していない SIP UA (User Agent) との接続性にも最大限配慮した標準とする。

ことを目的に以下の規定を行う。

- RFC7245 に規定されるセッション録音のアーキテクチャに関する事項
- RFC7866 に規定されるセッション録音における SIP シーケンス、SIP のハンドリング方法、RTP (Real-time Transport Protocol) のハンドリング方法およびメタデータの用法に関する事項

1.3 本報告書の内容

本仕様本文では、主として以下の事項について規定を行う。

- SRS と SRC は、B2BUA で接続されることを前提とし、SIP 網を介した SRS に接続するための動作規定として、SIP およびメタデータによる RS の確立、RTP の送受信時の動作について規定する。
- メッセージのフォーマットに関する規定を行う。
- シーケンス例を記載する。

1.3.1 本報告書の策定の背景と位置づけについて

本報告書作成時点において、日本国内ではセッション録音に関する規定がなかった。このため、本報告書策定においては、本報告書に準拠する SRS および SRC が、RS を確立し録音の実行が可能となるような共通の技術標準を策定することを目的として策定を行った。

1.3.2 本報告書の位置づけ

本報告書は、SRS および端末などの SRC が、RS を確立し録音を実行可能となるような技術基準として定められる。なお、本報告書の規定については、国内における SIP を利用し音声録音を行うことを目的としたサーバや端末における最低限の基準仕様として参照されることが期待される。

1.4 用語

用語	意味
セッション録音サーバ Session Recording Server (SRS)	SRS は、専用のメディアサーバもしくは録音対象のメディアを記録し保存を行う役割をもつ SIP UA である。SRS は典型的なマルチポートデバイスとして実装され、複数の送信元から送信された、複数のメディアを同時に受信する能力を持つ。また、SRS は録音されたセッションのメタデータも保存を行う。
セッション録音クライアント Session Recording Client (SRC)	SRC は、録音メディアの送信元の SIP UA であり、録音メディアを SRS に送信する役割を持つ。SRC は論理的な機能である。この能力は、ひとつまたは複数の物理デバイスで構成される。実際には、SIP 端末などの個人端末や SIP メディアゲートウェイ、セッション・ボーダー・コントローラ、SIP メディアサーバなどのアプリケーションサーバが SRC となることができる。SRC はメタデータを SRS に提供する。
コミュニケーションセッション Communication Session (CS)	ふたつもしくはそれ以上の SIP UA 間で確立されるセッションで、録音の対象となる。
録音セッション Recording Session (RS)	SRC と SRS の間で確立される録音を目的とした CS。
Recording-aware ユーザエージェント Recording-aware User Agent (Recording-aware UA)	CS に関連付けられた SIP 拡張により、セッションが録音されていることを認識可能な SIP UA。このような拡張は Recording-aware UA にセッションが録音されていることを通知する目的で使用される。もしくは、Recording-aware UA が録音をスタート、ストップ、ポーズ、再開するために使用される。
Recording-unaware ユーザエージェント Recording-unaware User Agent (Recording-unaware UA)	CS に関連付けられた SIP 拡張により、セッションが録音されていることを認識できない SIP UA。Recording-unaware UA は、録音されることを通知された場合や録音のスタート、ストップ、ポーズ再開を通知されても認識することができず、SIP 録音アーキテクチャの対象外となる。
録音メタデータ Recording Metadata	メタデータは SRS で必要となる CS の情報を記述している。例えば、CS に参加しているユーザの識別子とダイアログの状態を含む。典型的には、メタデータは複製メディアと一緒に SRS でアーカイブされる。メタデータはリアルタイムで SRS

	に通知される。
複製メディア Replicated Media	SRCによって作成されるCSに関連し、SRSに送信されるメディアのコピー。複製メディアはCSに関連するすべてのメディア(例えばオーディオやビデオなど)もしくはその一部(オーディオのみなど)を含む。複製メディアはRSの一部である。

2. セッション録音アーキテクチャ (RFC7245 第3章)

2.1 SRC の役割

この節では、セッション録音において SRC が、どのように物理的な構成を分類し、論理的な機能を提供しているのか、アーキテクチャを例示する。

2.1.1 B2BUA を SRC にする場合

B2BUA は、SRC としてメディアの録音を行うことができる。B2BUA は、CS の初回確立時にセッションを録音する必要があることを認識しているか、もしくは、セッションを確立したあとにセッションを録音することを決定する。

SRC が RS の開始を決定した場合、INVITE リクエストを SRS に送信する。

SRS が RS の開始を決定した場合、INVITE リクエストを SRC に送信することにより、RS を確立する。

RS の INVITE は、そのセッションが録音する目的で確立されたセッションであることを識別するための情報や、SRC から RS が開始される場合に、セッションが誤って SRS ではない UA にルーティングされてしまうことを防ぐための情報を含んでいる。

B2BUA/SRC は、録音される CS に関する UA に通知を行う責任がある。

B2BUA/SRC は、録音することを知らされた UA からのリクエストや、CS が録音すべきではないことを示すポリシーに従う責任がある。

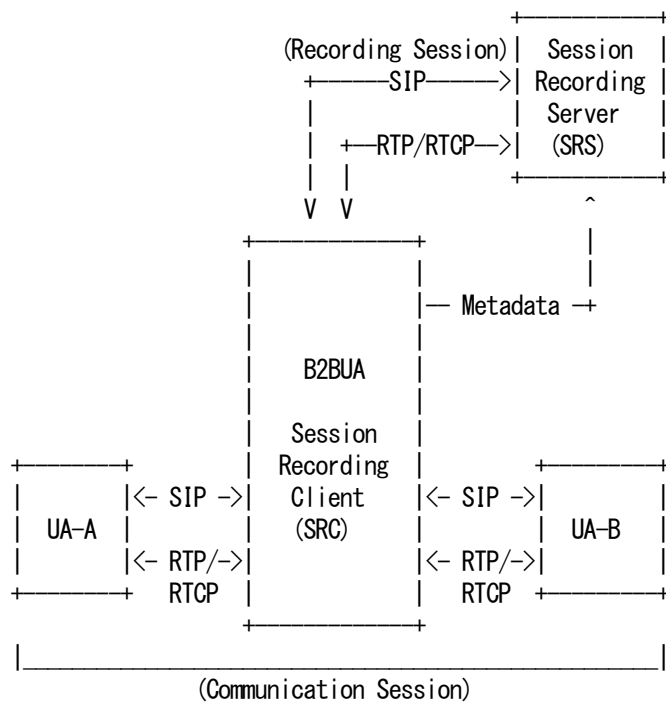


図 2.1 : B2BUA を SRC にする場合

2.1.2 エンドポイントを SRC にする場合

SIP エンドポイント / UA は、SRC として動作することができる。この場合、エンドポイントはメディアを複製し、SRS に送信する。

エンドポイントが RS の開始を決定した場合、SRS に INVITE を送信することで、SIP セッションを確立する。

SRS が RS の開始を決定した場合、INVITE リクエストをエンドポイントに送信することにより、SIP セッションを確立する。実際に RS を開始するかどうかを決めるメカニズムについては、SIP メディア録音のアーキテクチャでは定義されない。

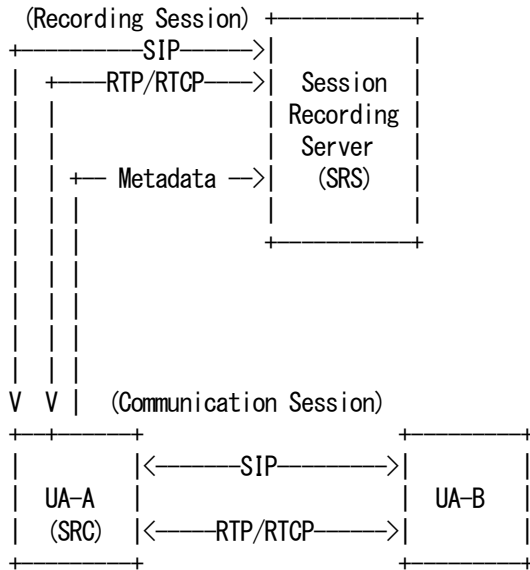


図 2.2 : エンドポイントを SRC にする場合

2.1.3 SIP プロキシは SRC には不適

SIP プロキシは、メディアにアクセスすることがなく、複製されたメディアを SRS に送ることができないため、SRC となることはできない。

2.2 RS の確立

SRC や SRS は、RS を開始できる。

RS は、SIP メッセージのレベルでも、セッションのレベルでも、CS とは独立している。

メディアのネゴシエーションについては、一般的な SIP / SDP (Session Description Protocol) の能力が使用されるべきであり、存在するトランスコーディングやメディアの暗号化は、排除されるべきではない。

2.2.1 SRC による録音の開始

SRC は、SRS にメディアを送信するために、下記の手順で、RS を開始する。

- SRC は、SRS の URI (Unified Resource Identifier) を準備する。URI は、通常 [RFC3263] の手順で解決される。
 - SRC は、SRS に INVITE リクエストを送ることで、ダイアログを初期化する。ダイアログは、[RFC3261] で定義されている通常の手順によって確立される。
 - INVITE メッセージには、確立された RS がメディアを録音する目的を含む。
 - SRC は、それぞれのメディアラインにおいて、即座にメディアの複製を開始する場合、SDP の属性に “a=sendonly” を含んでおく。また、メディアを送信する準備ができていない場合は、“a=inactive” を含んでおく。
 - SRC は、録音を行うために、メディアストリームを複製し、SRS に送信する。
- RS は、すべての CS に関連するメディア、もしくは、一部を複製する。

2.2.2 SRS による録音の開始

SRS は、下記の手順で、SRC との RS を開始する。

- SRS は、SRC の URI を準備する。URI は、通常 [RFC3263] の手順で解決される。
- SRS は、SRC に INVITE リクエストを送る。
- INVITE メッセージには、確立された RS がメディアを録音する目的を含む。
- SRS は、録音されているそれぞれのセッションを識別する。セッションを識別する仕組みは SRC のポリシーに依存する。
- SRS は、それぞれのメディアラインにおいて、即座にメディアの複製を開始する場合、SDP の属性に “a=recvonly” を含んでおく。

また、メディアを受信する準備ができていない場合は、“a=inactive”を含んでおく。

SRS が、どのメディアストリームを録音可能かという情報を事前に持っていない場合、Offerless INVITE を使用することにより、SRC に初回の SDP Offer を送信することを許可する。

2.2.3 RS の一時停止/再開

SRC および SRS は、SDP の Direction 属性を “inactive” に変更することで、録音の一時停止することができる。また Direction 属性を “recvonly” または “sendonly” に変更することで、再開することができる。

2.2.4 メディアストリームのミキシング

オーディオのみで構成される基本的なセッションの場合、ふたつの UA の間で双方向に典型的なふたつの RTP ストリームが存在する。このようなメディアを録音する場合、SRC から SRS にメディアを送信する時点で、ふたつのストリームをミックスする場合としない場合がある。

ミックスしない場合は、別々のストリームとして SRS に送信される。またこの場合は、別々のストリームとして送信されることを、必ず SDP で SRS に通知しなければならない [MUST]。

ミックスする場合は、ひとつのミックスされたストリームが SRS に送信される。

2.2.5 メディアのトランスコード

SRC が、CS と RS の間でメディアのトランスコードを行うことにより、これらのセッションはそれぞれ別々の標準的な SDP オファー/アンサーモデルによってネゴシエートされる。もし SRC がメディアのトランスコードをする能力がない場合、SRS に送信するメディアフォーマットは、CS でネゴシエートされたメディアに制限されるか、または、事前に SRS がサポートするメディアフォーマットを通知しておき、CS ではそのメディアフォーマットでオファーを送信する。しかし、一般的に SRS は、SRC よりもメディアフォーマットの対応範囲が広く、また、SRC は、[RFC5369] に示されるようなメディアのトランスコードを行うことが可能である。

2.2.6 損失のない録音

セッション録音は、一定のコミュニケーション環境の中で行われることが要件として挙げられる。このような環境は、一般的に“損失のない録音”として知られており、このような要求が課されることがある。

全体を通して損失のない録音を実現するためには、複数のレイヤーにおいて施策を行うことが必要である。個々の施策としては、適切な QoS を保証されたネットワークであることや、録音されるメディアの伝送を信頼性の高い方法を使用すること、あるいは、SRC と SRS において一定の SIP-REC のプロトコルレベルを満たした能力を持つ、といったことが挙げられる。

3. 録音シーケンス (RFC7866 第5章)

この章は、参考情報として、録音オペレーションの説明を提供する。

4章は、SRCとSRSとの間のRSにおけるSIP通信と、CSに参加するrecording-aware UAのための手順を記述する。

5章は、RSにおけるSDP (Session Description Protocol) の取扱いと、録音表示、選択の手順を記述する。

6章は、RSでのRTP処理を記述する。

7章は、SRCからSRSへ録音メタデータを配信するメカニズムについて記述する。

アーキテクチャ文書 [RFC7245] で述べられているように、SRCのロケーションに基づいて、いくつかの種類のコールフローがある。

3.1節で説明するサンプルコールフローは、SRCとSRS間の操作の概要を示す。

3.1 記録メディアの配信

SRC機能を持つSIP B2BUAが、UA AからUA Bの呼をルーティングする場合、SRCはUA間のメディアパスにアクセスできる。

SRCが会話を録音する必要があることを認識すると、SRCはB2BUAにUA AとUA Bの間でメディアを中継させる可能性があります。

次に、SRCはSRSとRSを確立し、複製されたメディアをSRSに向けて送信する。

また、エンドポイントはSRC機能を有することができ、その場合はエンドポイント自体がSRSへのRSを確立する。

エンドポイントはCS内のメディアにアクセスできるため、エンドポイントは複製されたメディアをSRSに送信できます。

図3.1および図3.2におけるコールフロー例は、SRSに向かってRSを確立するSRCを示す。

図3.1は、SRCとして動作するUA Aを示す。

図3.2は、SRCとして動作するB2BUAを示す。

CSが確立された後で、SRCがRS (図3.2のメッセージ (5)) を確立していることを示すコールフローを提示しているが、SRCはCSとは無関係にRSを確立するタイミングを選択することができることに注意すること。

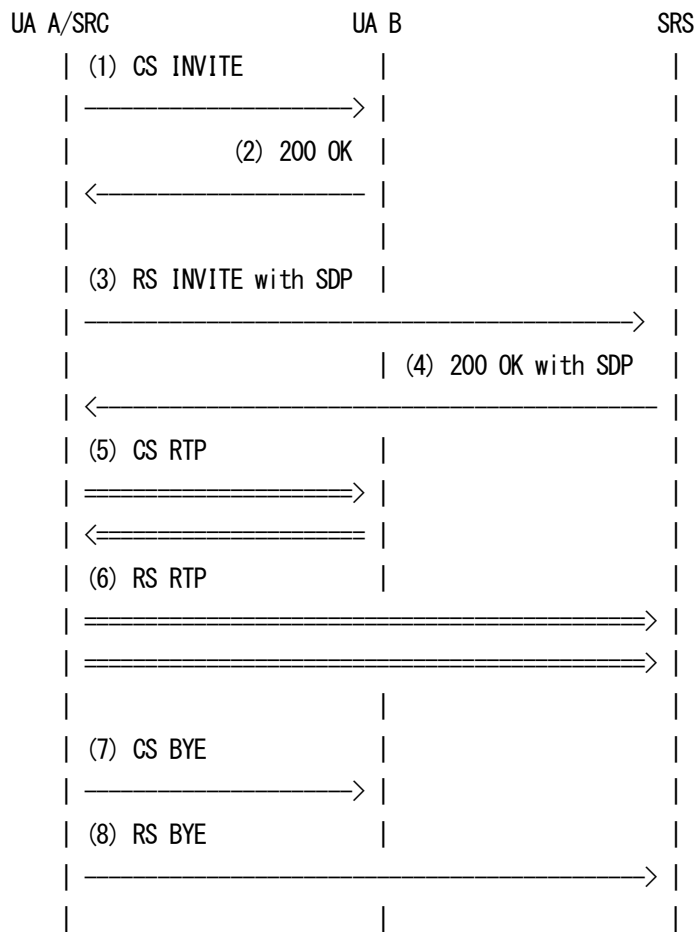


図 3.1 : SRC が UA となる基本的な録音コールフロー

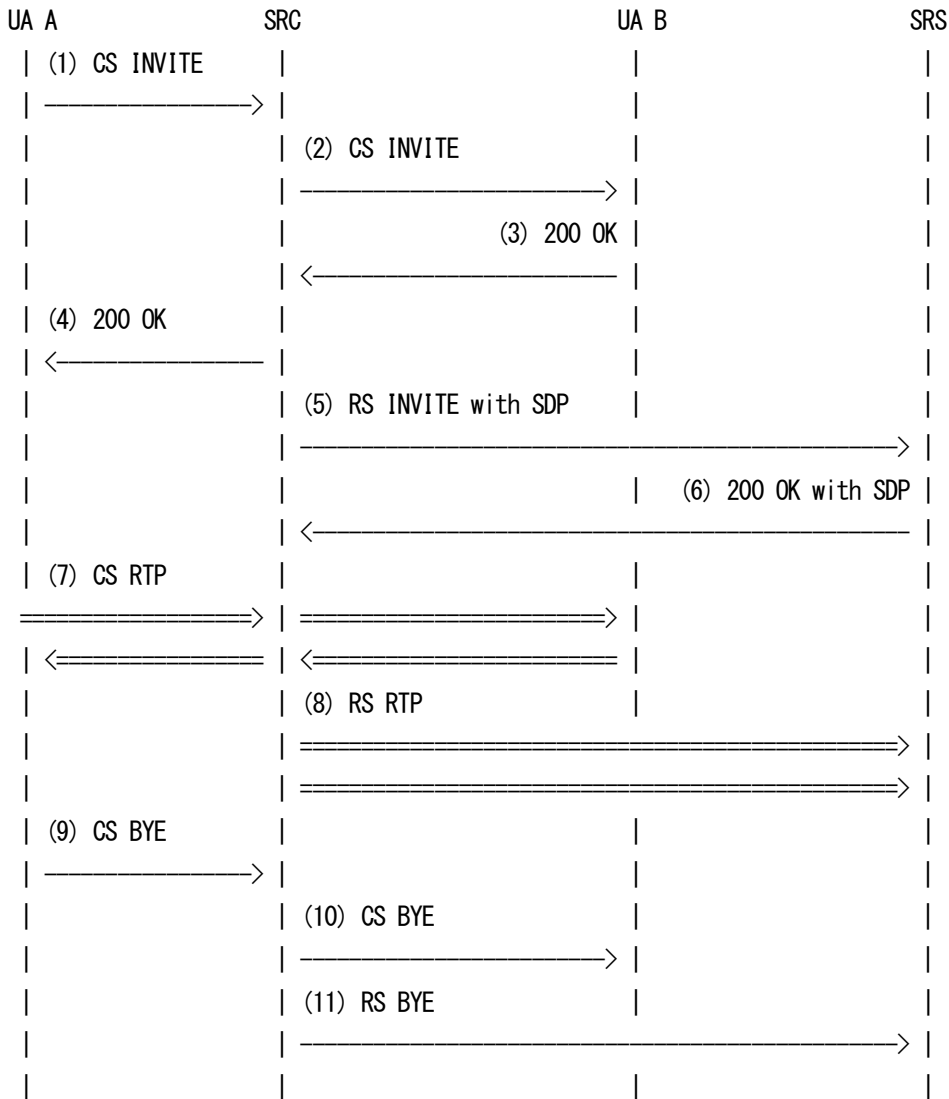


図 3.2 : SRC が B2BUA となる基本録音コールフロー

図 3.2 に示すコールフローは、ミキサーを使用した集中型会議の場合にも適用できる。

明確にするため、INVITE に対する ACK、および BYE に対する 200 OK は示されていない。

会議の中心は、各会議参加者からのすべてのメディアにアクセスできるため、SRC 機能を提供できる。

録音が要求されると、SRC はメタデータとメディアストリームを SRS に配信する。

会議の中心はミキサーにアクセスできるため、SRC はすべての参加者からのメディアストリームをミックスし、SRS への 1 つのミックスメディアストリームとすることを選択することがある。

SRC は、複数の CS を録音するために、単一の RS を使うことができる。

SRC が新しい通話を録音することを望むたびに、SRC は RS に新しい録音するストリームを追加するために新しい SDP オファーで RS を更新する。これに対応して新しいコールのメタデータも更新する。

SRS は SRC に RS を確立することもできる、しかし、SRS が録音するための通話を指定する方法を定めることはこの文書の範囲外である。

3.2 録音メタデータの配信

SRC は、SRS へのメタデータの配信を担当する。

SRC は、RS 内のイニシャル INVITE 内容の録音するメディアストリームに関する初期メタデータスナップショットを提供することができる。

以降のメタデータ更新は SRC が送信した UPDATE または re-INVITE リクエストのイベントのストリームとして表すことができる。

これらのメタデータの更新は通常、初期メタデータスナップショットの増分更新であり、更新のサイズを最適化する。ただし、SRC は、いつでも新しいメタデータスナップショットを送信することを決定する場合がある。メタデータは、INVITE メッセージまたは UPDATE メッセージのボディで伝送される。録音するメディアストリームの属性などの特定のメタデータは、RS の SDP 内に配置される。SRS には、SRC に新しいメタデータスナップショット更新を求める要求を送信する機能がある。これは、何らかの理由で SRS が現在の増分更新ストリームを理解できない場合に発生する。たとえば、SRS が内部障害によって現在の状態を失った場合など。SRS は、オプションでスナップショットリクエストと共に理由を付加することができる。この要求により、SRC と SRS の両方が状態を新しいメタデータスナップショットと同期させることができるため、以降の増分メタデータ更新はその最新のメタデータスナップショットに基づいて行われる。メタデータ・コンテンツと同様に、メタデータスナップショット要求は、RS の SRS によって送信された UPDATE または INVITE メッセージのコンテンツとして転送される。

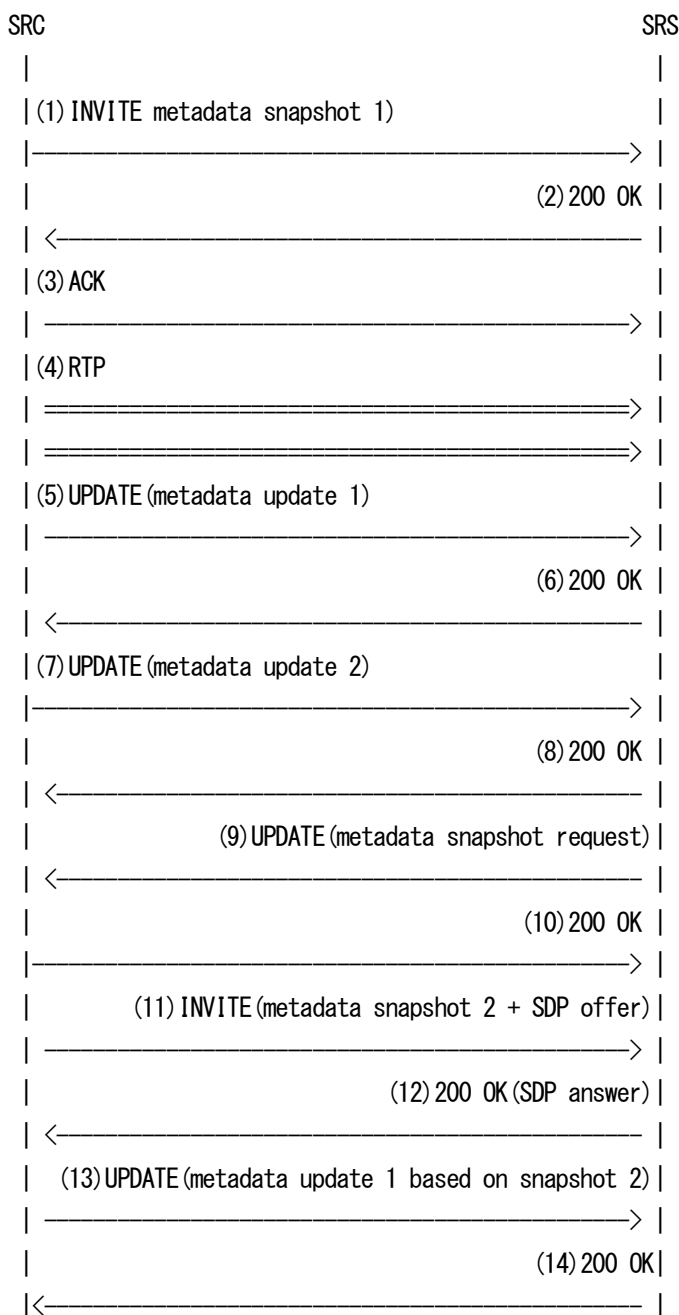


図 3.3 : SIP UPDATE によるメタデータ配信

3.3 録音表示の受信と録音の提供

初期設定

SRC は、CS の参加者に録音指示を提供する責任がある。

recording-aware UA は、SDP 「a=record」属性によって録音指示を受信することをサポートし、SDP 「a=recordpref」属性を含むことによって、CS 内に録音設定を指定することができる。

録音属性は、録音が行われているかどうかを示す CS 内の SRC による宣言である。

録音設定属性は、その録音設定を示すための CS 内の recording-aware UA による宣言である。

録音されたくない UA には、録音が行われていることが通知されることがある (たとえば、その設定の表示の能力が無く、その設定が無視された)。

これが発生した場合、録音されることを避ける UA の唯一のメカニズムは、セッションへの参加を終了することである。

属性がどのように使われるかについて説明するために、UA A が UA B に呼び出しを始めている、そして、UA A が録音を実行している SRC でもあるならば、UA A が a=record:on で SDP オファーの録音指示を提供する。UA A が SRC であるので、UA A は直接 SRC から録音指示を受け取る。UA B が SDP 申し込みを受けるとき、UA B は録音がこのセッションの他のエンドポイントで起こっているのを見る。UA B が SRC ではなく、録音設定も提供しないので、SDP アンサーには a=record または a=recordpref を含まない。

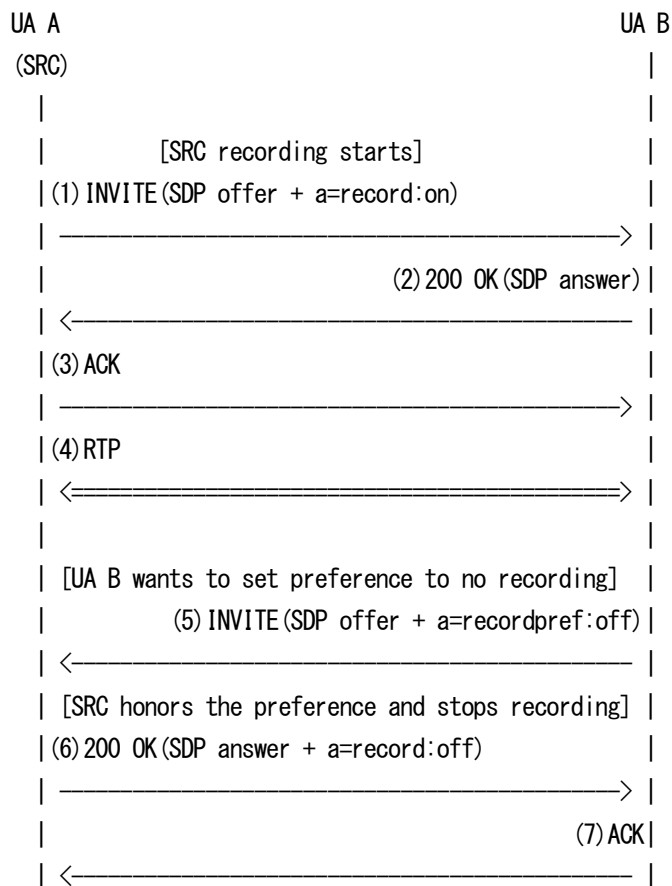


図 3.4 : 録音指示と録音設定

通話が確立され、録音中が進行中の場合、UA B は後で録音設定を録音無しに変更することを決定すると、「a=recordpref」属性で re-INVITE を送信する。選択を守ることは SRC 次第である、そして、この場合、SRC は記録を止めることに決定し、SDP アンサーで録音指示を更新する。

UA B が (2) (最初の INVITE のための 200 の OK) で録音設定を明示的に示すことができたことに注意すべきである。INVITE に最初の INVITE または最初の応答において録音の選択を示さないことは、第一に録音されているユーザのチャンスを減らすかもしれない。

4. SIP ハンドリング (RFC7866 第6章)

4.1 SRC における手順

4.1.1 RS の開始

RS は特別に拡張され適用された SIP セッションであり、この拡張は SRC と SRS のために以下の手順で記載される。SRC 又は SRS が RS でない SIP を受ける時、SIP セッションで何をしたらよいかを決定するのは SRC 又は SRS 次第である。SRC は、SIP INVITE 要求を SRS に送ることによって RS を開始できる。SRC と SRS は From と To ヘッダでそれぞれ特定される。SRC は Contact URI に "+sip.src" 機能タグを含む必要がある [MUST]、このことは全ての RS に対して [RFC3840] への拡張として仕様で定義されている。RS の目的として生成されるやり取りを決定するために、SRS は要求と応答を生成・変更するやり取りの中で "+sip.src" 機能タグの存在を使用する。付け加えて、SRC が REGISTER 要求を送るとき、SRC は SRC であることを示すために "+sip.src" 機能タグを含んでもよい [MAY]。SIP Caller Preferences 拡張子は、プロキシをルーティングする実装として任意であるので、RS が SRC か SRS に送られるという保証が全くない。新しいオプションタグ "siprec" が導入される。RFC3261 に従って、SRC か SRS だけが RS でこのオプションタグを受け入れることができる。セッション録音プロトコルの拡張をサポートしない UA が 420 (Bad Extension) 応答で単に INVITE 要求を拒絶できるように RS を開始するときには、SRC は要求ヘッダに "siprec" オプションタグを含む [MUST]。SRC が新しい INVITE を受信するとき、"+sip.srs" 機能タグと "siprec" オプションタグの両方が INVITE 要求に含まれている場合、SRC は SIP セッションが RS であると判断する [MUST]。

4.1.2 録音の指示と設定のための SIP 拡張

CS として、SRC は CS に関係するすべてに録音指示を提供する [MUST]。CS 内で関係する UA は、"record-aware" オプションタグを提供することによって recording-aware であることを示すことができ、SRC は以下の5章で説明する SDP "a=record" 属性の新しい SDP を提供する [MUST]。"record-aware" オプションタグが無い場合 -- 関係する UA が recording-aware でないことを意味する -- SRC は、インバンドトーンの再生もしくは適所で符号付きの関係する約定を持つ等、他の方法で録音識別を提供する [MUST]。CS 内の SRC は "+sip.src" 機能タグを含むことで SRC であることを暗示してもよい [MAY]。recording-aware の参加者は SRC が CS にあり、5章で記述される新しい SDP 「a=recordpref」属性記述により、録音設定を CS に設定することができる。

4.2 SRS における手順

SRS が新しい INVITE を受信したとき、INVITE 要求に "+sip.src" 機能タグと "siprec" オプションタグの両方が含まれる場合は、SRS は RS としてみなす [MUST]。SRS は SRC に SIP INVITE 要求を送信することで RS を始める。SRS と SRC は、それぞれ From と To ヘッダで識別される。SRS は、全ての RS では [RFC3840] として Contact URI に "+sip.srs" 機能タグを含む [MUST]。SRC は、RS ([RFC6341] 内の REQ-030) を目的としてやり取りが生成されたことを確認するために、やり取りの生成と変更要求と応答において機能タグの存在を使用する。付け加えて、SRS が REGISTER 要求を送信するときに、SRS はそれが SRS であることを示すために "+sip.srs" 機能タグを含む [SHOULD]。SRS は RS を開始するとき、[RFC3261] として Require ヘッダに "siprec" オプションタグを含み [MUST]、それによりセッション録音プロトコルをサポートしない UA は、単に 420 (Bad Extension) で INVITE 要求を拒否する。

4.3 Recording-aware UA のための手順

recording-aware UA は、録音識別を受信し、通話に対して録音設定を要求するために録音 SIP と SDP 拡張をサポートする CS における参加者である。recording-aware UA は、CS の開始または確立する際、新しい "record-aware" オプションタグを持つ SRC によって提供される録音識別の通知を受け付けることができることを示し [MUST]、このことは initial INVITE 要求または応答の Supported ヘッダに "record-aware" オプションタグを含むことを意味する。recording-aware UA は、録音が on か off か、on/off の中間として停止されているか、適切なユーザーインターフェースを通して利用者に対して録音識別を提供する [MUST]。適切なユーザーインターフェースはリアルタイムの通知もしくは装置の使用で録音を受け付けられるかといった事前に確立された取り決めを含んでもよい [MAY]。自動装置であるいくつかの UA (例えば、IVR、メディアサーバ、PSTN ゲートウェイなど) は、録音識別を表すためのユーザーインターフェースは持っていないか、そのような UA が録音識別を指示するときは、UA は PSTN ゲートウェイ上のインバンドトーン、ログファイル内に録音識別の出力、VoiceXML やり取りでのアプリケーションイベントの出力、といった他の方法で識別を表す [SHOULD]。これらの UA は録音識別を指示することを選択しなくてもよく [MAY]、それによって、録音を意味するためにインバンドトーンの再生のように SRC が選択するどんなメカニズムかに頼る。

5. SDP ハンドリング (RFC7866 第7章)

5.1 SRC における手順

SRC と SRS は [RFC3264] に記述された SDP のオファー/アンサーモデルに従う。SRC と SRS のための手順は、RS で使用される協定を記述する。

5.1.1 RS における SDP ハンドリング

SRC は SRS からメディアを受信することを想定していないため、SRC は一般的に [RFC3264] の手順に従って、"a=sendonly" 属性をメディアストリームに付与することにより、メディアを送るためだけに SDP オファーの各メディアストリームをセットする。SRC は SRS に参加者の録音されたストリーム情報を送り、SRC は、残りのメタデータがある録音されたストリームを識別するために各メディアストリーム上に [RFC4574] として、"label" 属性 ("a=label") を提供する [MUST]。"a=label" 属性は各メディアストリームを識別し、[RFC7865] としてラベル名はメタデータ内のメディアストリーム参照にマッピングされる。"a=label" 属性の範囲は、SDP およびラベルとして表される SIP 要求または応答のメッセージボディの中で伝達されたメタデータに適用されるだけである。録音ストリームは CS ストリームとは別であることに注意；メタデータは、各録音ストリームを与える参加者のリストを提供する。

図 5.1 は、ストリームが録音されたオーディオとビデオの両方を持つ SRC を提供する SDP の例を示す。この例は 72 文字より長い拡張行を含んでいることに注意。；これらの行は <allOneLine> タグの間で取得される。

```
v=0
o=SRC 2890844526 2890844526 IN IP4 198.51.100.1
s=-
c=IN IP4 198.51.100.1
t=00
m=audio 12240 RTP/AVP 0 4 8
a=sendonly
a=label:1
m=video 22456 RTP/AVP 98
a=rtptime:98 H264/90000
<allOneLine>
a=fmtp:98 profile-level-id=42A01E;
      sprop-parameter-sets=ZOIACpZTBYmL,aMljiA=
</allOneLine>
a=sendonly
a=label:2
m=audio 12242 RTP/AVP 0 4 8
a=sendonly
a=label:3
m=video 22458 RTP/AVP 98
a=rtptime:98 H264/90000
<allOneLine>
a=fmtp:98 profile-level-id=42A01E;
      sprop-parameter-sets=ZOIACpZTBYmL,aMljiA=
</allOneLine>
a=sendonly
a=label:4
```

図 5.1 : オーディオとビデオのストリームを持つ SRC を提供する SDP の例

5.1.1.1 メディアストリーム更新の手順

RS の発生～終了までを通して、SRC は様々な理由のために RS に録音ストリームの追加と削除ができる -- 例えば、CS のストリームが CS に加えられた、削除された、もしくは RS が複数の CS を扱うのであれば CS が生成又は終了したなど。RS から録音されたストリームを削除するためには、[RFC3264] の手順によれば、SRC はポートをゼロにしてストリームが削除された新しい SDP オファーを送る。RS に録音されたストリームを追加するためには、[RFC3264] の手順によると SRC は新しいメディアストリーム記述を追加するか以前に無効にされた旧メディアストリームを再利用することにより、新しい SDP オファーを送る。SRC は、ミュート録音のような理由で SRC から SRS への録音メディアのストリーミングと収集を一時的に中止することができる。この場合、[RFC3264] の手順に従って SRC は新しい SDP オファーを送り、一時停止される各々の録音ストリームのために非アクティブ (`a=inactive`) をメディアストリームにセットする。ストリーミングと録音メディアの収集を再開するために、SRC は新しい SDP オファーを送り、送信のみ (`a=sendonly`) をメディアストリームにセットする。SDP を更新することにより、CS は自身でメディアストリームの方向を変更してもよいことに注意、-- 例えば、保留の SDP として `a=inactive` をセットする。メディアストリームの方向は、SRC によってメタデータで伝達される。CS のメディアストリームが、`inactive` へ又は `inactive` から変化する時、対応する RS への効果として SRC のポリシーで管理される。SRC は、対応する CS のメディアストリームが `inactive` であるときに RS のメディアストリームを一時停止するローカルポリシーを有してもよく [MAY]、又は SRC は `sendonly` として RS のメディアストリームを切り離してもよい [MAY]。

5.1.2 CS における録音識別

CS が録音されていることの識別を提供するためのメカニズムが存在する間、そのメカニズムは通常、インバンドトーンの再生もしくは参加者への通知のように、CS メディアストリームに提供される。SRC が CS 内で `recording-aware` UA に録音状態を示すことを許容するために、新しい "レコード" SDP 属性が使用される。"レコード" SDP 属性は、SDP オファー/アンサーのどちらかにおいて、メディアレベル又はセッションレベルとして現れる。属性がセッションレベルで適用されるとき、指示は SDP のすべてのメディアストリームに適用される。属性がメディアレベルで適用されるとき、指示はその1つのメディアストリームだけに適用され、また、セッションレベルで設定される場合は指示が上書きされる。録音の終了のように、録音識別が変更を必要とするときはいつでも、SRC は `re-INVITE` または "`a=record`" 属性の SDP を更新するための UPDATE を実行する [MUST]。

以下は "`record`" 属性の ABNF (Augmented Backus-Naur Form) [RFC5234] である。

```
attribute = / record-attr  
; RFC 4566 で定義された属性
```

```
record-attr = "record:" indication  
indication = "on" / "off" / "paused"
```

`on`: 録音は実行中である。
`off`: 実行中の録音は無い。
`paused`: 録音は実行中であるが、メディアは一時停止中である。

5.1.3 CS での録音優先

SRC が SDP オファー/アンサーにおいて "`a=recordpref`" SDP を受信したとき、SRC はローカルポリシーでの録音優先を選択する。もし、SRC が録音状態を変更するのであれば、SRC は SDP アンサー又はその後の SDP オファーにおいて "`a=record`" 属性で録音状態を報告する [MUST]。

5.2 SRS における手順

標準的に、SRS は SRC から RTP ストリームを受信するだけであり、従って、[RFC3264] の手順によれば、"`a=recvonly`"属性により、SRS からの SDP オファー/アンサーは一般的に受信したメディアに各メディアストリームに設定する。SRS が録音されたストリームを受信する準備ができていないとき、[RFC3264] の手順によれば、"`a=inactive`"属性により、SRS は SDP オファーまたはアンサーの中に `inactive` としてメディアストリームを設定する。SRS が録音されたストリームを受信する準備ができたとき、SRS は "`a=recvonly`" 属性でメディ

アストリームを設定した新しいSDP オファーを送る。図 5.2 は、図 5.1 の SDP オファーに対する SRS からの SDP アンサーの例を示す。この例は 72 文字より長い展開された行を含むことに注意；これらの行は <allOneLine> タグの間で取得される。

```
v=0
o=SRS 0 0 IN IP4 198.51.100.20
s=-
c=IN IP4 198.51.100.20
t=0 0
m=audio 10000 RTP/AVP 0
a=recvonly
a=label:1
m=video 10002 RTP/AVP 98
a=rtpmap:98 H264/90000
<allOneLine>
a=fmtp:98 profile-level-id=42A01E;
    sprop-parameter-sets=ZOIACpZTBYmI,aMljiA==
</allOneLine>
a=recvonly
a=label:2
m=audio 10004 RTP/AVP 0
a=recvonly
a=label:3
m=video 10006 RTP/AVP 98
a=rtpmap:98 H264/90000
<allOneLine>
a=fmtp:98 profile-level-id=42A01E;
    sprop-parameter-sets=ZOIACpZTBYmI,aMljiA==
</allOneLine>
a=recvonly
a=label:4
```

図 5.2 : オーディオとビデオストリームを持った SRS からの SDP アンサー例

RS のライフタイムを超えると、SRS は様々な理由で RS から録音されたストリームを削除することができる。RS から録音されたストリームを削除するために、[RFC3264] の手順によると、SRS はメディアストリームのポートを削除するようにゼロを設定した新しい SDP オファーを送る。SRS は、SRS が新しい SDP オファーを送る時に RS 内に録音されたストリームを追加してはならない [MUST NOT]。同様に、SRS が RS を開始するとき、録音されたストリームを持った SDP オファーを SRC が生成させるために、SRS は SDP オファー無し の INVITE を開始する [MUST]。図 5.3 にあるシーケンス図は、SRS が最初は録音されたストリームを受信するための準備がなく、SRS が録音の準備ができたときに RS を更新する例を示す。

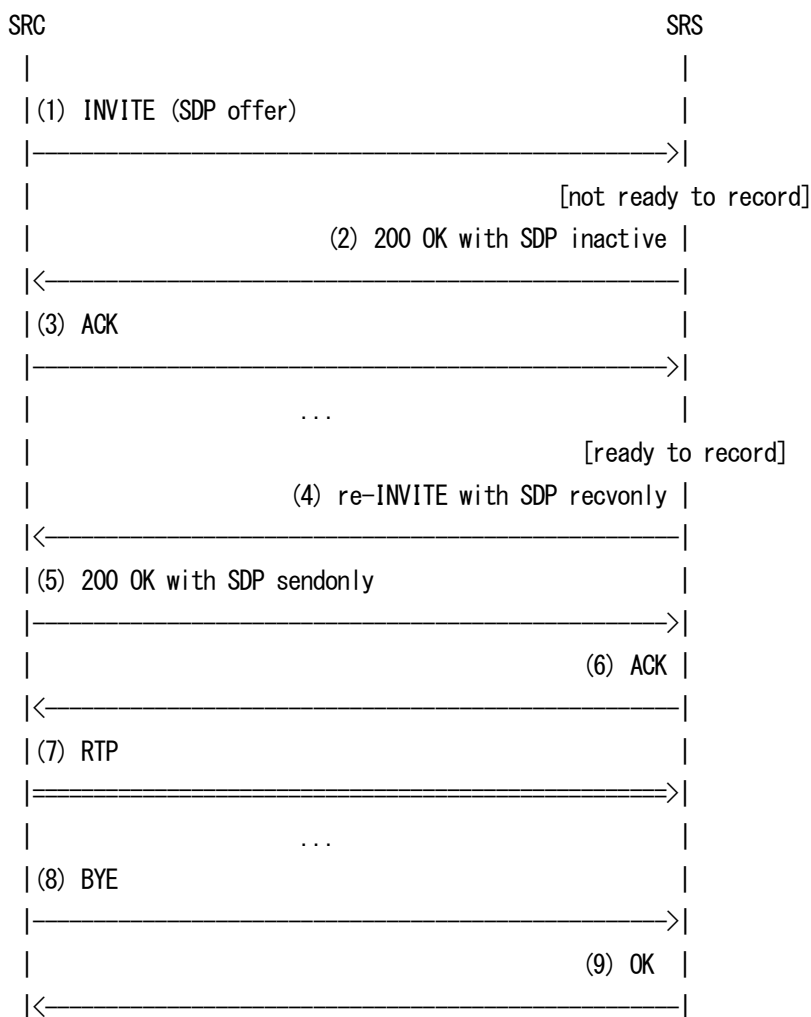


図 5.3 : a=inactive を持った要求に対する SRS の応答

5.3 Recording-Aware UA のための手順

5.3.1 録音識別

recording-aware UA が "a=record" 属性を含む SDP オファーまたはアンサーを受信するとき、UA はそれぞれの媒体について、直近に受信した "a=record" SDP 属性に基づいてエンドユーザーに録音それぞれの媒体に on/off/一時停止であるかの識別子を提供する。CS が B2BUA 又は会議集中のような複数の UA を経由して横断するとき、CS が録音されていることを認識している CS に関わる各 UA は、CS 内の全ての相手に "a=record" 属性により録音識別を提供する [MUST]。同じ CS の通話パス内に複数の SRC が存在することができるが、録音識別属性はどんな SRC であるかとか、いくつ SRC があるかといったヒントを提供しない。エンドポイントは、通話が録音されていることのみ認識する。更に、特定の SRC のために録音を開始又は停止するための要求としてこの属性は使われない。

5.3.2 録音選択

CS の参加者は、セッション確立もしくはセッション中の録音または未録音を設定するのに CS 内で録音選択を設定してもよい [MAY]。

新しい "recordpref" SDP 属性が導入され、CS の参加者は、セッション確立時もしくはセッション中にどの SDP オファー/アンサー内にこの録音選択属性を設定してもよい。SRC は SRC におけるローカルポリシーに基づいて、参加者からの録音選択を要求しない。また、参加者は 5.3.1 項で記述された "a=record" SDP 属性を通して録音識別を認識することができる。SDP の "a=recordpref" 属性はメディアレベルもしくはセッションレベルで表示することができ、SDP オファー/アンサーに表示することもできる。属性がセッションレベルで適用される場合、録音選択は SDP 内の全てのメディアストリームに適用される。属性がメディアレベルで適用される場合、録音選択は 1 つのメディアストリームのみ適用し、セッションレベルにも設定される場合には録音選択は上書きされる。UA は、その後の SDP オファーまたはアンサー内で "a=recordpref" 属性に変えることで録音選択を変更する。SDP 内に "a=recordpref" 属性が存在しないことは、UA

が録音選択を持っていないことを意味する。以下は "recordpref" 属性の ABNF である。

```
attribute =/ recordpref-attr
```

; RFC 4566 で定義された属性

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

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

on: 未だ開始されていない場合、録音する選択が設定される。現在録音が一時停止している場合は、選択は録音の再開になる。

off: 録音しない選択が設定される。現在録音を開始している場合は、選択は録音の停止になる。

Pause: 録音が現在実行中である場合、選択は録音の一時停止になる。

nopreference: UA は録音に関する設定は無いことを表す。

6. RTP ハンドリング (RFC7866 第 8 章)

この章は、SIPREC [RFC6341] の文中で RTCP (Real-time Transport Control Protocol) に関する推奨とガイドラインを提供する。最も効果的に通信するために、SRC と SRS と recording-aware UA は、明確に予測できる方法で RTP により提供されたメカニズムを利用すべきである。これらのメカニズムを意識させることと推奨とガイドラインを提供することは、このドキュメントの目標である。

6.1 RTP メカニズム

この節は、重要な RTP/RTCP 構造と SIPREC の文中で特に役立つメカニズムについて簡潔に説明する。

6.1.1 RTCP

RTP データ転送は、RTCP によって増加させられて、データ送信のモニタリングを許容する。RFC3550 で定義される RTCP は、RTP セッションの中ですべての参加者への制御パケットの周期的なトランスミッションに基づいて、データ・パケットと同じ分配メカニズムを使用する。RFC3550 では、RTCP のサポートは必要であり [REQUIRED]、特に提供される。SIPREC に関連した以下の重要な機能性:

1) データ分配の品質のフィードバック

受信者からのフィードバックは、分配における欠点を分析するのに使用してもよい。そういうものとして RTCP はメディアの受付に関して起こる問題について記録されることになっており、SRS が SRC に知らせるために SRC が recording-aware UA に知らせるために明確で効率的なメカニズムである。

2) RTP ソースに対して持続する転送レベル識別子 (CNAME、または標準的な名前) を含んでいる

同期ソース (synchronization source (SSRC)) RFC3550 の識別子は、衝突が発見されるかプログラムが再開される場合は変更してもよい。その場合、受信者が各参加者の動向をおさえるために CNAME を使用できる。また、受信者は関連する RTP セッションの 1 セットの特定の参加者から複数のデータ・ストリームを関連づけるのに CNAME を使用してもよい。 -- 例えばオーディオとビデオを同期させるように。また、メディアストリームの同期は、データ送信者の RTCP パケットに含まれる NTP (Network Time Protocol) と RTP のタイムスタンプによって容易になる。

6.1.2 RTP プロフィール

暗号化された RTP ストリームを使用するとき、SRC、SRS、および recording-aware UA のための RECOMMENDED RTP プロフィールは「RTCP ベースのフィードバック (RTP/SAVPF (Secure Audio-Visual Profile with Feedback)) として拡張的で安全な RTP プロフィール」 RFC5124 である。そして、暗号化されないメディアストリームを使用するとき「RTCP ベースのフィードバック (RTP/AVPF (Audio-Visual Profile with Feedback)) として拡張的な RTCP プロフィール」 RFC4585 である。しかしながら、これらは必要条件でないの、いくつかの実装では「SRTP」 RFC3711 と「オーディオおよび最小限のビデオ会議の RTP プロフィール」 RFC3551 を使用してもよい。したがって、SRC、SRS、および recording-aware UA が RTP/SAVP と RTP/AVP を使用することで、部分的に達成可能な中心機能のために RTP/SAVPF か RTP/AVPF を完全に当てにしない [RECOMMENDED]。AVPF と SAVPF は、厳密な帯域幅に従うよりもイベントに対応して RTCP パケットのフレキシブルなトランスミッションを許容する改良された RTCP タイマモデルを提供する。VPF ベースのコーデックコントロールメッセージは、場面変化、エラー回復、ダイナミックな帯域幅調整などのイベントを扱うために、SRC、SRS、および recording-aware UA に効率的なメカニズムを提供する。

SAVP と SAVPF は、メディア暗号化、完全性保護、再生保護、およびソース認証の限定フォームを提供する。それらは特定の認証メカニズムを含まず、必要としない。

6.1.3 SSRC

RFC3550 で定義される SSRC は、RTP ヘッダと RTCP パケットのフィールドに入る。無作為の 32 ビットの番号が、RTP セッション内でグローバルなユニークになるために必要である。番号が慎重に選ばれているのは重要で、参加者が同じネットワークにいるか、同時に同じ番号を選ばないようにするためである。SSRC 値の選択と衝突解決に関するガイドラインを RFC3550 に提供する。また、SSRC は一つの RTP セッション内でメディアのさまざまな原因を切り分ける為に使用してもよい。衝突解決の良い理由として、SRC、SRS、および recording-aware UA ハンドルは、SSRC 値の変更理由を適切に特定して変更する。RTCP 内に入る CNAME 値は、この識別を容易にする。

6.1.4 CSRC

RFC3550 で定義される役立つソース (Contributing Source (CSRC)) は、RTP ミキサーによって発生した結合ストリームに提供した RTP パケットストリームを特定する。ミキサーは、特定の packets 世代に提供したソースに関する SSRC 識別子のリストをその packets の RTP ヘッダに挿入する。このリストは CSRC リストと呼ばれる。ミキサーとして機能するとき、SRC か recording-aware UA がそれらに従って CSRC リストを設定して、それを受信する時に SRC と SRS が RFC3550 の CSRC リストを解釈する [RECOMMENDED]。

6.1.5 SDES

RFC3550 で定義されるソース記述 (Source Description (SDES)) は、ゼロのリストがあとに続いた SSRC/CSRC 識別子か、SSRC/CSRC の情報を配送するよりも多くの項目を含んでいる。エンドシステムは、それら自身のソース識別子 (SSRC と固定 RTP ヘッダで同じ) を含む 1 つの SDES パケットを送る。そのような 31 以上のソースがあれば、ミキサーはそれが SDES 情報、または複数の完全な SDES パケットを受信している各 CSRC の塊を含む 1 つの SDES パケットを送る。個々の CSRC を特定する能力は SIPREC の文中で重要である。メタデータ RFC7865 は、シグナリングレベルでこれを達成するためのメカニズムを提供する。SDES は RTP レベルでメカニズムを提供する。

6.1.5.1 CNAME

RFC3550 で定義される標準的なエンドポイント識別子 (Canonical End-Point Identifier (CNAME)) は、SSRC 識別子から一定のままのソース (送信側または受信側) 識別子までの結合を提供する。SRC と recording-aware UA が CNAME を適切に発生させて、SRC と SRS がそれらを解釈して使用するのには重要である。CNAME 値を発生させるためのガイドラインは、RFC7022 という「RTP 制御プロトコル (RTCP) の標準的な名前 (Canonical Names (CNAMEs)) を選ぶためのガイドライン」で提供する。

6.1.6 Keepalive

SIPREC のメディアストリームが、多くの正当な理由で延長期間に非活性状態で存在するかもしれないことが予期される。NAT/ファイアウォールでの結合と他のどのようなピンホールもそのような間隔の間に活性状態のままであるため、SRC、SRS、および recording-aware UA は、すべての RTP メディアストリームで「RTP/RTP 制御プロトコル (RTCP) に関連している NAT マッピングを生かすためのアプリケーションメカニズム」RFC6263 で推奨された keepalive 手順に従う [RECOMMENDED]。

6.1.7 RTCP フィードバックメッセージ

「RTP Audio-Visual Profile with Feedback (AVPF) のコーデック制御メッセージ」RFC5104 は、AVPF RFC4585 で定義されたメッセージの拡張を指定する。これらのメッセージのサポートと適切な用法は、SRC、SRS、および recording-aware UA の実現に重要である。AVPF か SAVPF RTP プロフィールを使用するときだけ、これらのメッセージが適切であることに注意すべきである。

6.1.7.1 完全なイントラリクエスト

Full Intra Request (FIR) コマンドは、指定されたメディア送信者により受信すると、メディア送信者ができるだけ早い機会にデコーダリフレッシュポイントを送信することを必要とする。デコーダリフレッシュポイントの使用は、ストリーム中のその後の画像送信のコード化プロセスの参照時にポイントの前で画像送信の使用を控えることを含む。デコーダリフレッシュポイント (H.264 映像コーデックのイントラか Instantaneous Decoding Refresh (IDR) の画像) が、予測した画像サイズより大きいことが一般的に有り得る。したがって、有効なビット伝送速度がわずかであるシナリオでは、デコーダリフレッシュポイントの使用は典型的な画像の持続時間よりかなり長い遅れを含む。

6.1.7.1.1 FIR の代わりに SIP INFO の推奨しない用法

「メディア制御のための XML スキーマ」RFC5168 は、ビデオ高速更新のために拡張マークアップ言語 (Extensible Markup Language (XML)) スキーマを定義する。下位互換性の目的以外の RFC5168 で説明されたメソッドを使用するので、推奨できない。実際には、代わりに FIR メッセージを使用する [SOULD]。一般的なメカニズムが SRC と SRS の間に存在するのを確実にするために、SRS は両方のメカニズム (FIR と SIP INFO) をサポートする [MUST]。SRC とネゴシエートする時は FIR メッセージを使用して、そうでない場合

は SIP INFO を使用する。

6.1.7.2 Picture Loss Indication

RFC4585 で定義される Picture Loss Indication (PLI) は、1 枚以上の画像に属する未定義の符号化映像データロスのエンコーダを知らせる。RFC4585 は、エラーを回復する FIR メッセージの代わりに PLI を使用することを勧める。FIR は、ユーザに対してビデオを使用不可能にするデコーダリフレッシュポイントを送らない状況だけに適切である。メッセージを FIR に送るのが適切な例は、新しいユーザが会議に参加して、一定のデコーダリフレッシュポイント間隔が確立されないときのマルチポイント会議、およびストリームを変更するビデオを切り換える Multipoint Control Unit (MCU) を含んでいる。PLI と FIR の適切な使用は、最小のオーバーヘッドで撮影して保存したビデオが確実に使用可能になるために重要である (例えば、プレーヤーが撮影して保存したビデオをレンダリングするように、必要な基準座標系は存在している)。

6.1.7.3 Temporary Maximum Media Stream Bit Rate Request

受信側、トランスレータ、またはミキサーが、メディアストリームのために最大のビット伝送速度を提供された値に制限するように送信側に要求するために、Temporary Maximum Media Stream Bit Rate Request (TMMBR) RFC5104 を使用する。TMMBR の適切な使用は、利用可能な帯域幅に変更するのにすばやく適合することを容易にする。

6.1.7.3.1 SDP 帯域幅属性の再ネゴシエーション

TMMBR によって示された新しい値が残りのセッションで有効であるのなら、TMMBR 送付側がセッションシグナリングプロトコルを使用することで、セッション上限の再ネゴシエートを実行すると予想される。したがって、SIPREC に関して、実際は帯域幅の一時的変更と再ネゴシエートにより恒久的な変更として、SDP オファー/アンサーで TMMBR を使用する [RECOMMENDED] 。

6.1.8 送信と受信に関するシンメトリック RTP/RTCP

SDP オファー/アンサー交換の中では、RTP エンティティはパケットを受信するための RTP と RTCP 転送ポートアドレス (すなわち、IP アドレスとポート番号) を選ぶ。パケットを送信するとき、RTP エンティティはパケットを受信するために指示されたものと同じソースポートか、異なるソースポートを使用してもよい。RTP の送信と受信で使用される転送ポートアドレスが同じであるとき、それは「シンメトリック RTP」RFC4961 と呼ばれる。RTP を送信するとき、シンメトリック RTP の使用は必要である [REQUIRED]。RTCP を送信するとき、シンメトリック RTP の使用は必要である [REQUIRED]。SRS は通常 RTP を送信しないが、RTCP を送信して RTP と RTCP を受信する。同様に、SRC は SRS から通常は RTP を受信しないが、RTCP を受信して RTP と RTCP を送信する。

注意: シンメトリック RTP/RTCP は、RTP/RTCP マルチプレクシング RFC5761 とは異なる。

6.2 役割

SRC は、1 つ以上の CS と情報を転送する様々な UA から対応する RS の文中の SRS までの集合メディアに関するタスクを持っている。SRC がこれを実行してよい多数の方法があり、限定はしないが CS の中の UA として現れるか、CS の中の UA との間の B2BUA として現れることを含む。

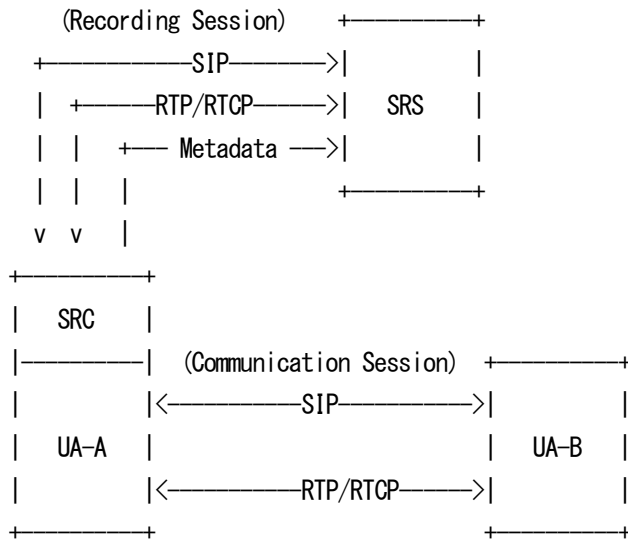


図 6.1 : SRC としての UA

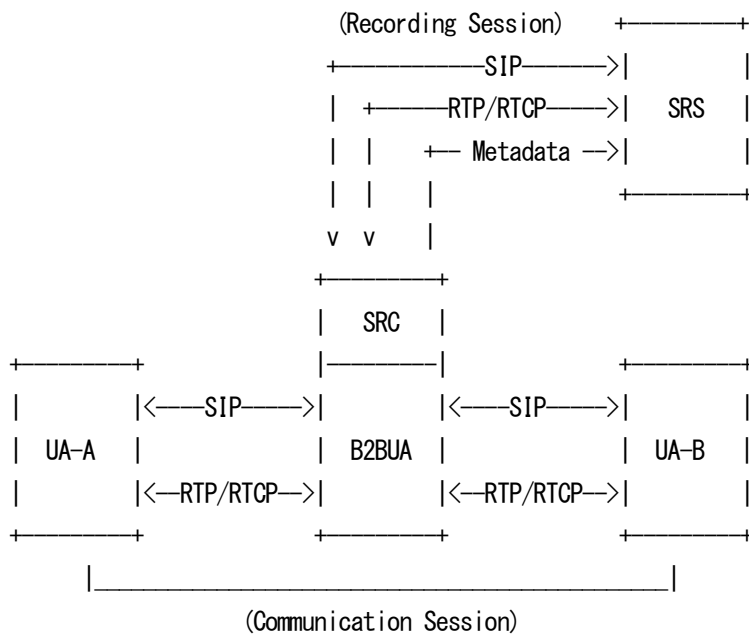


図 6.2 : SRC としての B2BUA

以下の項は、CS 中の UA に関する見解、および RS 中の SRS に基づいて、SRC が再生を選択してよい役割のセットを定義する。CS と対応する RS は独立しているセッションである。したがって、SRC は対応する RS の中で再生するよりも CS の中で異なる役割を再生してもよい。

6.2.1 RTP トランスレータとして機能する SRC

SRC は、RFC3550 で定義されるようにトランスレータとして振舞ってよい。トランスレータの定義の特性は、SSRC 識別子はそのままで RTP パケットを転送するという点である。2つのタイプのトランスレータがある: それ単に転送するものと、転送に加えてコード変換を実行するものである (例えば、1つのコーデックから別のものまで)。

6.2.1.1 転送トランスレータ

転送トランスレータとして振舞うとき、さまざまな原因 (例えば、異なった SSRC と異なった UA) から別々のストリームを受信した

RTP は、SRC によってミックスできずに別々に SRS に送信する [MUST]。UA と SRS がどのような SSRC 衝突も検出できるように、UA と SRS の間の SRC はすべての RTCP レポートを透過する [MUST]。ストリームを送信する UA によって発生した RTCP Sender Reports は、SRS に転送する [MUST]。SRS で発生した RTCP Receiver Reports は、関連する UA に転送する [MUST]。UA は、RTCP Receiver Reports の複数のセットを受信してもよい – 他の UA から CS に参加する 1 つ以上か、SRS から RS に参加する 1 つである。UA が認識している状態で録音しているなら、UA は SRS から RTCP Receiver Reports を処理する [SHOULD]。SRTP が CS と RS の両方で使用されるなら、復号化、そして/または、再暗号化が発生してもよい。例えば、異なったキーが使用されていると、それは発生する。同じキーが使用されているなら、発生する必要はない。パケットロスが、UA から SRC まで、または、SRC から SRS までに発生するなら、SRS はロスを検出して回復を試みる [SHOULD]。関連 RTP と RTCP にパケットを送る以外には、SRC はこれに対する役割を果たさない。

6.2.1.2 コード変換トランスレータ

コード変換トランスレータが振舞うとき、SRC はコード変換 (例えば、1 つのコーデックから別のものまで) を実行してよく [MAY]、SRC が CS で受けるものと SRC が RS に送るもの間のパケットの異なったレートであってもよい。転送トランスレータが振舞うとき、別々のストリームとしてさまざまな原因 (例えば、異なった SSRC と異なった UA) から受信した RTP は、SRC によってミックスさせることができず、SRS へ別々に送信する [MUST]。UA と SRS がどんな SSRC 衝突も検出できるように UA と SRS の間の SRC はすべての RTCP レポートを透過する [MUST]。ストリームを送信する UA によって発生した RTCP Sender Reports を SRS に転送する [MUST]。SRS で発生した RTCP Receiver Reports は、関連する UA に転送される [MUST]。SRC は、どのようなコード変換も考慮に入れるために RTCP Receiver Reports の操作を必要としてもよい。UA は、RTCP Receiver Reports の複数のセットを受信してもよい – 他の UA から CS に参加する 1 つ以上、および、SRS からの RS に参加する 1 つ。recording-aware UA は、SRS から RTCP Receiver Reports を処理するように準備して [SHOULD]、recording-unaware UA はそのような RTCP パケットは無関係であり捨ててもよい。SRTP が CS と RS の両方で使用されるなら、復号化、そして/または、再暗号化が発生してもよい。例えば、異なったキーが使用されていると、それは発生する。同じキーが使用されているなら、それは発生する必要はない。パケットロスが、UA から SRC まで、または、SRC から SRS までに発生するなら、SRS はロスを検出して回復するのを試みる [SHOULD]。関連 RTP と RTCP にパケットを送る以外に、SRC はこれにおける役割を果たさない。

6.2.2 RTP ミキサーとして機能する SRC

RFC3550 で定義されるように、SRC が RTP ミキサーとして機能する場合は、SRC は異なった UA から RTP ストリームを合成して、SSRC を使用する SRS に向けてストリームを送信する。提供している UA から SSRC は、CSRC 識別子がストリーミングされるのに従って運ばれる [SHOULD]。SRC は、受信したストリームの中でタイミング調整をしても良く、SRS にストリームを送信するときに自身のタイミングを発生させてよい。任意にミキサーとして機能する SRC はコード変換を実行できて、異なる UA から受信した異なるコーディングに対処できる。RTCP Sender Reports と Receiver Reports は、ミキサーとして機能する SRC によって転送されないが、RTCP Source Description (SDS) パケットを転送する必要がある。SRC は、関連する UA と SRS に向かってそれ自身の RTCP Sender Reports と Receiver Reports を発生させる。RS に対する SRC と SRS の間の SRTP の使用は、CS に対する UA と SRC の間の SRTP の使用からは独立している。UA から SRC までにパケットロスが発生するなら、SRC はロスの検出と回復を試みる [SHOULD]。SRC から SRS までにパケットロスが発生するなら、SRS はロスの検出と回復を試みる [SHOULD]。

6.2.3 RTP エンドポイントとして機能する SRC

RFC3550 で定義されるように、SRC が RTP エンドポイントとして機能するケースは、ミキサーケースと同様である。SRC と SRS の間の RTP セッションが、CS の一部である RTP セッションから完全に独立していると考えられることを除く。SRC は必要ではないが、SRS に送信する前に異なる参加者からの RTP ストリームをミックスすることができる。SRC と SRS の間の RTCP は、CS 上の RTCP から完全に独立している。SRC と SRS の間の SRTP の RS の使用は、UA と SRC の間の SRTP の CS の使用から独立している。UA から SRC までにパケットロスが発生するなら、SRC はロスの検出と回復を試みる [SHOULD]。SRC から SRS までにパケットロスが発生するなら、SRS はロスの検出と回復を試みる [SHOULD]。

6.3 SRC による RTP セッション用法

SRC には、SRS へ録音メディアを転送するのに選択して良い複数の方法がある。いくつかの場合は、RS の中のすべてのメディアにただ一つの RTP セッションを使用して良いが、他の場合は、複数の RTP セッションを使用して良い。以下の項は、SRC での基本的な RTP

セッション用法の例を提供する。以前、どのような RTP 構造とメカニズムが使用されていたかに関する議論を含んでいる。SRC は一つの RS の中で RTP セッション用法の 1 つ以上を使用するのを選択してもよい。SRC と SRS、SRC の間のベース相互運用性の目的としては、1 つの CS メディア方向につき、SDP に別々の m 行をサポートする [MUST]。RTP セッション用法が説明するセットが、網羅していることは示さない。

6.3.1 複数の m 行を使用する SRC

複数の m 行を使用するとき、SRC は、SRS への SDP オファーに各 m 行を含む。SRS からの SDP アンサーは、すべての m 行を含む [MUST]。RFC3264 でポートが示されなくても、拒否された m 行でも。あるいは、SRC がアンサーで示されたサポート水準が容認できないと考えるなら、それは代替の RTP セッション用法がネゴシエートされる別の SDP オファー/アンサー交換を引き起こしてもよい。RS の CS の中にメディアに関するマッピングを参加者として保存するために、SRC は CS の中のそれぞれのユニークな CNAME を RS の中のユニークな CNAME にマップする [SHOULD]。さらに、SRC は CS の中で CNAME/SSRC のそれぞれのユニークな組み合わせを RS の中のユニークな CNAME/SSRC にマップする [SHOULD]。その際に、RTP トランスレータとして、または、RTP 終点として SRC は振舞ってよい。

図 6.3 は、各 UA がそれぞれ一つの SSRC と共に 2 つの RTP セッション (例えば、オーディオとビデオ) を提供する参加者の代理となる場合である。SRC は RTP トランスレータとして振舞い、それぞれ一つの SSRC と共に 4 つの RTP セッションを使用することでメディアを SRS に渡す。それらメディアストリームの中で UA によって使用された CNAME と SSRC 値は、SRC から SRS までのメディアストリームに保存される。

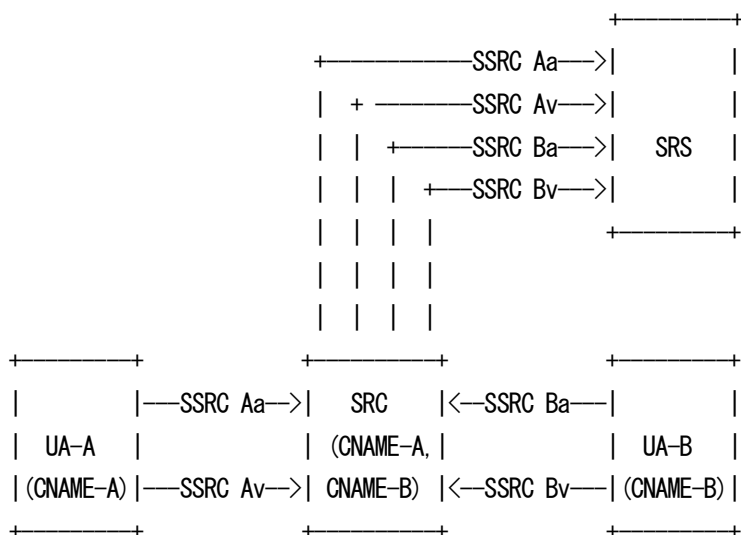


図 6.3 : 複数の m 行を使用する SRC

6.3.2 ミキシングを使用する SRC

ミキシングを使用するとき、SRC は異なる参加者から RTP ストリームを結合して、その SSRC を使用して SRS に向けて結合した RTP ストリームを送信する。提供している参加者からの SSRC は、CSRC 識別子として運ばれる [SHOULD]。SRC は、SDP オファーにおけるそれぞれの RTP セッションあたり 1 つの m 行を SRS に含める。SRS からの SDP アンサーは、すべての m 行を含める [MUST]。RFC3264 でポートが示されなくても、拒否された m 行でも。アンサーを受信すると、SRC はアンサーで示すにみられるようにメディアを SRS に送り始める。RS において、CS の中の参加者へのメディアに関するマッピングを保存するために、SRC は CS の中のそれぞれのユニークな CNAME を RS の中のユニークな CNAME にマップする [SHOULD]。さらに、SRC は CS の中で CNAME/SSRC のそれぞれのユニークな組み合わせを RS の中のユニークな CNAME/SSRC にマップする [SHOULD]。CSRC として RS で使用される時は、必要であれば SSRC を書き直して、SRC は SSRC 衝突を避ける [MUST]。その際に、SRC は RTP ミキサーとして機能する。SRS が CSRC 値の用法をサポートしない場合、それぞれのミックスされたストリームの中に含まれる参加者を決定するために SIPREC メタデータに依存する。図 6.4 は、各 UA が、それぞれ一つの SSRC と共に 2 つの RTP セッション (例えば、オーディオとビデオ) を提供する参加者の代理となる場合である。SRC は、RTP ミキサーとして機能して一つの SSRC と 2 つの CSRC を含む一つの RTP セッションに各参加者からのメディアをミックスした 2 つの RTP セッションを使用することでメディアを SRS に渡す。

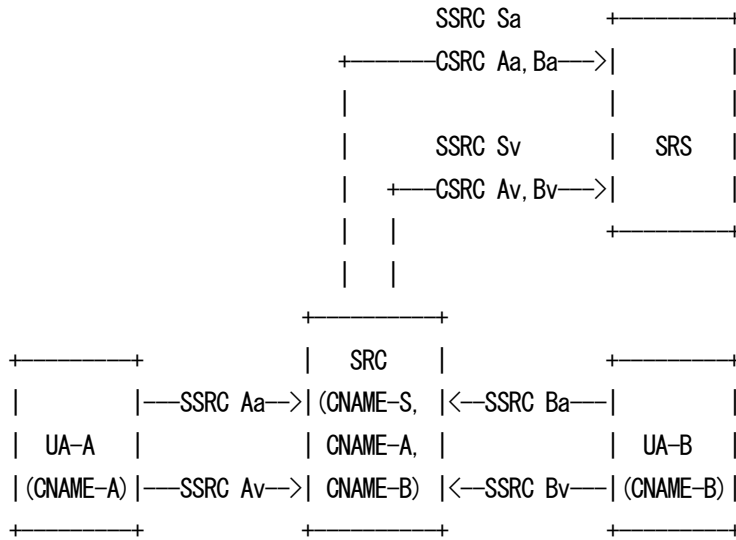


図 6.4 : ミキシングを使用する SRC

6.4 SRS による RTP セッション用法

オーディオ CS の録音をサポートする SRS は、SDP (CS メディアあたり 1 つ) で、別々のオーディオ m 行の SRC 用法をサポートする [MUST]。ビデオ CS の録音をサポートする SRS は、SDP (CS メディア指示あたり 1 つ) で、別々のビデオ m 行の SRC 用法をサポートする [MUST]。したがって、典型的なオーディオ通話をサポートする SRS に関して、SRS は少なくとも 2 つのオーディオ m 行の受信をサポートしなければならない。典型的なオーディオとビデオ通話をサポートする SRS に関して、SRS は SDP の少なくとも 4 つ全ての m 行の受信をサポートしなければならない。- 2 つのオーディオ m 行と 2 つのビデオ m 行。これらの要件では、オーディオ録音に関するサポートは必要とせずにビデオのみをサポートして実装する SRS を許容する。また、CS で 1 つのストリームの 1 方向のみの録音をサポートして実装する SRS を許容する。- 例えば、オーディオなしでビデオを送るだけ (受信しない) の監視カメラを録音するために設計された SRS。これらの要件は、一つの m 行を使用して別々のオーディオストリームを一緒にミックスするように、他のモードが実装されて使用されるのを防ぐために記載されない。むしろ、要件は相互運用性のために実装する一般的なベースモードを提供するために記載される。カメラやプレゼンテーションのように、参加者がビデオ通話で 1 つ以上の m 行をサポートするなら、一般的なベースモードをサポートする SRS の実装が CS のすべてのメディアストリームを録音しなくて良い事に注意するのは重要である。SRS の実装は、他のモードをサポートするかもしれない。それらが基本的な相互運用性に関する一般的なベースモードで共同利用するように、上で議論したモードを少なくともサポートしなければならない。

7. メタデータ (RFC7866 第9章)

一部のメタデータ属性は SDP に含まれ、その他の属性は "application/rs-metadata" という新しいコンテンツタイプに含まれている。メタデータのフォーマットは [RFC7865] のメカニズムの一部として記述されている。メタデータを運ぶ目的で、Content-Disposition の新しい "disposition-type" が定義されている。

値は "recording-session" であり、これは "application/rs-metadata" コンテンツが SRS によって処理されるメタデータを含んでいることを示す。

7.1 SRC の手順

SRC は、RS での SRS にメタデータを送信しなければならない [MUST]。SRC は、メタデータを利用可能になるとすぐに、またメタデータが変更されるたびに送信すべきである [SHOULD]。

SRC がメタデータを送信する前に一時的に待つことを正当化できるケースは次のとおり。

- 以前のメタデータ交換が完了するのを待つ (すなわち、SRC は、以前のオファー/アンサーが完了するまで別の SDP オファーを送信することができず、この間に UPDATE を送信したくない場合もある)。
- RS 上でシグナリングレートを強制する。
- メタデータに影響を与えるすべてのイベントではなく、キーイベントが発生したときにメタデータを送信する。

また、SRC は、プライバシーに関する懸念から特定のメタデータを抑制するように構成されてもよいし、録音にそれを含める必要性がないと認識してもよい。

SRC によって送信されたメタデータは、フルメタデータスナップショットまたは部分的な更新のいずれかに分類される。フルメタデータスナップショットは、RS に関連するすべてのメタデータを記述する。

SRC は、いつでもフルメタデータスナップショットを送信することができる [MAY]。

SRC は、フルメタデータスナップショットが以前に送信された場合にのみ、部分的な更新を送信することができる [MAY]。

SRC は、INVITE 要求、UPDATE 要求 [RFC3311]、または SRS からのオファー無し INVITE への 200 レスポンスのメタデータ (フルメタデータスナップショットまたは部分的更新のいずれか) を送信することができる [MAY]。

メタデータに SDP ラベルへの参照が含まれる場合、メタデータを含む要求には、それらのラベルを定義する SDP オファーも含まなければならない。

SIP メッセージに SDP オファーとメタデータの両方が含まれている場合、要求本体は、SDP オファーを含む 1 つの下位ボディ部分とメタデータを含むもう 1 つのコンテンツタイプ "multipart/mixed" を持つ必要がある [MUST]。

SIP メッセージに SDP オファーまたはメタデータのみが含まれる場合、"multipart/mixed" コンテナはオプションである。

SRC は、RS を確立する最初の INVITE リクエストにフルメタデータスナップショットを含めるべきである [SHOULD]。

メタデータがまだ利用可能でない場合 (例えば、CS が存在しない場合に RS が確立された場合)、SRC はメタデータが利用可能になるとすぐにフルメタデータスナップショットを送信すべきである [SHOULD]。

SRC が SRS からスナップショット要求を受信した場合、SRC はすぐにフルメタデータスナップショットを送信する必要がある [MUST]。

図 7.1 は、最初の INVITE リクエストで SRC によって送信されたフルメタデータスナップショットの例を示しています。

```
INVITE sip:recorder@example.com SIP/2.0
Via: SIP/2.0/TCP src.example.com;branch=z9hG4bKdf6b622b648d9
From: <sip:2000@example.com>;tag=35e195d2-947d-4585-946f-09839247
To: <sip:recorder@example.com>
Call-ID: d253c800-b0d1ea39-4a7dd-3f0e20a
CSeq: 101 INVITE
Max-Forwards: 70
Require: siprec
Accept: application/sdp, application/rs-metadata
Contact: <sip:2000@src.example.com>;+sip.src
Content-Type: multipart/mixed;boundary=foobar
Content-Length: [length]
```

--foobar

Content-Type: application/sdp

v=0

o=SRS 2890844526 2890844526 IN IP4 198.51.100.1

s=-

c=IN IP4 198.51.100.1

t=0 0

m=audio 12240 RTP/AVP 0 4 8

a=sendonly

a=label:1

--foobar

Content-Type: application/rs-metadata

Content-Disposition: recording-session

[metadata content]

図 7.1 : RS のための INVITE リクエストのサンプル

7.2 SRS の手順

SRS は、SRC からのメタデータ更新を INVITE および UPDATE 要求で受信する。

SRC は以前の更新に基づいて部分的な更新を送信できるので、SRS は SRC からの更新のシーケンスを追跡する必要がある。

SRS で内部障害が発生した場合、SRS は SRC からの部分的な更新を認識できないことがある

SRS は、SRC にフルメタデータスナップショットを要求することによって、内部障害から回復することができる。

メタデータ情報の構文エラーやセマンティックエラーなどの特定のエラーは、SRC 側でエラーが発生している可能性が高く、フルメタデータスナップショットが要求されても同じエラーが再度発生する可能性がある。

同じエラーを繰り返さないようにするために、SRS は構文エラーまたはセマンティックエラーがメタデータ内で検出されたときに RS を単純に終了することができる。

(注) セマンティックエラー：意味の違いや矛盾によって生じるエラー

SRS は、UPDATE 要求を送信することによって、フルメタデータスナップショットを明示的に要求することができる [MAY]。

このリクエストは、Content-Disposition タイプ "recording-session" を持つボディを含む必要があり、SDP ボディを含んではならない

[MUST NOT]。

SRS は、UPDATE 応答または他の SIP トランザクションでフルメタデータスナップショットを要求してはならない [MUST NOT]。内容の形式は "application/rs-metadata" であり、本文は XML 文書であり、形式は [RFC7865] で定義されている。

図 7.2 に例を示す。

```
UPDATE sip:2000@src.example.com SIP/2.0
Via: SIP/2.0/UDP srs.example.com;branch=z9hG4bKdf6b622b648d9
To: <sip:2000@example.com>;tag=35e195d2-947d-4585-946f-098392474
From: <sip:recorder@example.com>;tag=1234567890
Call-ID: d253c800-b0d1ea39-4a7dd-3f0e20a
CSeq: 1 UPDATE
Max-Forwards: 70
Require: siprec
Contact: <sip:recorder@srs.example.com>;+sip.srs
Accept: application/sdp, application/rs-metadata
Content-Disposition: recording-session
Content-Type: application/rs-metadata
Content-Length: [length]
```

```
<?xml version="1.0" encoding="UTF-8"?>
  <requestsnapshot xmlns='um:ietf:params:xml:ns:recording:1'>
    <requestreason xml:lang="it">SRS internal error</requestreason>
  </requestsnapshot>
```

図 7.2 : メタデータ要求

SRS がメタデータスナップショットを要求するためには、ダイアログの状態に関係なく送信できるので、UPDATE が選択されていることに注意する。

これは、この操作のための UPDATE メッセージと re-INVITE メッセージの両方のサポートを必要とするよりも優れていた。

SRC がメタデータスナップショットの要求を受け取ると、別の INVITE または UPDATE トランザクションでフルメタデータスナップショットをただちに提供する必要がある [MUST]。

その後の部分的な更新は、このフルメタデータスナップショットの前に送信されたメタデータに依存しない。

SRS によって受信されたメタデータは、ある要素を別の要素に相互参照するために使用される ID 要素を含むことができる。

ID の定義とその ID への参照を含む要素は、しばしば同じ SRC から受信される。

これらの要素が異なる SRC から受信される場合も有効である。たとえば、同じ CS 内の各エンドポイントが SRC として機能し、共通 ID が同じ CS を参照する場合などです。

SRS はこれをエラーとみなしてはいけない [MUST NOT]。