セキュア メディアおよび SRTP パススルー

Cisco Unified Border Element（SP Edition）は、Secure Real-Time Protocol（SRTP）パススルーおよびセキュア メディアという 2 つの方式の暗号化データ ストリームをサポートしています。最適な方法は、SRTP パススルーを使用することで、エンド ポイント自体が暗号化機能を通知できるためです。

セキュア メディア機能は、すべてのコールに対してグローバル レベルでイネーブルで、デフォルトでディセーブルです。セキュア メディアがグローバルでオンになっている場合、SBC は、実際のエンド ポイント機能に関係なく、すべてのエンド ポイントが暗号化データ ストリームを使用することを前提としています。

Cisco IOS XE Release 2.6 より、シグナリングされないセキュア メディア機能を使用して、コール アドミッション制御（CAC）テーブル エントリ コマンドを使用し、特定のコールと隣接の詳細なレベルでセキュア メディアを設定できます。

CAC ポリシーを使用して、SRTP パススルーを詳細に設定できます。

暗号化メディア パケットを受け入れるように Cisco Unified Border Element（SP Edition）を設定するために使用される方式に関係なく、Cisco Unified Border Element（SP Edition）には、これらのパケットのパススルーを保証するために追加の帯域幅が確保されています。一般的に、メディア ストリームの帯域幅は、エンドポイントが使用するコーデックによって決定されます。ただし、メディア ストリームで暗号化を使用すると、パケット サイズが増加します。経験則として、帯域幅の要求は非暗号化コーデックよりも 10% 増加します。ただし、この増加は、メディア フロー統計に反映されません。

Cisco Unified Border Element（SP Edition）は、以前は Integrated Session Border Controller と呼ばれており、このマニュアルでは通常 Session Border Controller（SBC; セッション ボーダー コントローラ）と呼びます。

本章で使用されているコマンドの詳細な説明については、次の場所にある『*Cisco Unified Border Element (SP Edition) Command Reference: Unified Model*』を参照してください。

http://www.cisco.com/en/US/docs/ios/sbc/command/reference/sbcu_book.html

すべての Cisco IOS コマンドの詳細については、http://tools.cisco.com/Support/CLILookup にある Command Lookup Tool を使用するか、Cisco IOS マスター コマンド リストを参照してください。

セキュア メディアおよび SRTP パススルーの機能履歴

リリース	変更内容
Cisco IOS XE Release 2.4	これらの機能は、Cisco ASR 1000 シリーズの集約サービス ルータで導入されました。
Cisco IOS XE Release 2.6	CAC テーブル エントリ コマンドを使用して詳細なレベルで設定できるように、シグナリングされないセキュア メディア機能が導入されました。この機能の導入により、グローバル レベルのセキュア メディアの設定機能が廃止されました。

Cisco IOS XE Release 3.1S	SRTP と RTP 間のインターワーキングおよび SRTP パススルー機能が追加されました。
Cisco IOS XE Release 3.4S	RTP で多重化された RTCP と SSRC ベースの多重化の SRTP のサポート機能が追加されました。

内容

この章の内容は、次のとおりです。

セキュア メディアおよび SRTP パススルーの前提条件

両方の機能を実装するには、次の前提条件を満たす必要があります。

セキュア メディアおよび SRTP パススルー機能を実装する前に、Cisco Unified Border Element（SP Edition）がすでに設定されている必要があります。

セキュア メディアの制約事項

次に、グローバルおよびシグナリングされないセキュア メディアの制約事項を示します。

- この機能をイネーブルにすると、**show sbc dbe media-flow-stats** コマンドで表示される RTCP 関連統計が未知のものとして表示されます。

次に、シグナリングされない（詳細レベルの）セキュア メディアの制約事項を示します。

- コールの発信側と着信側の両方が、**caller secure-media** および **callee secure-media** コマンドで設定されている必要があります。コールの 1 つのレッグのみが設定されている場合、コールは失敗します。

（注） 状況によっては、**branch** コマンドを、**caller** または **callee** コマンドの代用として使用できます。**branch** コマンドはリリース 3.5.0 で導入されました。このコマンドの詳細については、「ダイレクト非制限 CAC ポリシーの設定」（P.139）を参照してください。

セキュア メディアについて

一般的に、エンドポイントは、メディア トラフィックが SIP シグナリングを通じて暗号化されていることを示します。暗号キーは、セッション記述プロトコル（SDP）を通じて交換されるか、Datagram Transport Layer Security（DTLS）メカニズムを使用して交換されます。

Cisco IOS XE Release 2.4 および Release 2.5 では、Cisco Unified Border Element（SP Edition）は暗号化されたメディア（DTLS または Secure RTP（SRTP））を使用するエンドポイントまたは SIP デバイスとインターワーキングするものの、エンドポイントはこれを SIP シグナリングで示しませんでした。これら以前のリリースでは、SBC はグローバルにイネーブルにしたセキュア メディアの設定をサポートしていましたが、SBC 上のすべてのコールは SRTP メディアで構成されるものとして扱われていました。エンド ポイントが SRTP メディアをシグナリングしない場合でも、メディア ピンホールは、トラフィックが SRTP であるかのように作成されました。SBE サブモードのグローバル コンフィギュレーションでは、エンドポイントが暗号化された SRTP メディアを使用していることを示していますが、そのように SIP シグナリングを使用して通信やネゴシエーションを行いません。グローバル レベルでこの設定が適用されると、暗号化されていないフローに対してであっても、追加の帯域幅が確保され、RTP および RTCP チェックと検証はディセーブルになります。

SRTP メディア ストリームのシグナリングを完全にサポートしていない SIP デバイスとインターワーキングする場合、SBC は、メディアが SRTP としてシグナリングされないため、メディアが SRTP であることを事前に確認できません。Cisco IOS XE Release 2.6 以降、シグナリングされないセキュア メディア機能を使用することで、SBC は、SRTP メディアを生成するもののこれを通常の RTP メディア ストリームとしてシグナリングする SIP デバイスと相互運用できます。

通信するどの SIP デバイスでシグナリングされない SRTP をサポートする必要があるかを知るように SBC を設定できます。そのような SIP デバイスは常に SRTP メディアを送信すると見なされます。最低でも、SRTP のサポートが必要な特定の隣接のすべてのデバイスを詳細に設定する必要があります。詳細なレベルでのセキュア メディアを設定する際、コール アドミッション制御（CAC）テーブル エントリ コマンドを使用します。セキュア メディアをグローバルに有効にするのではなく、セキュア メディアを使用するコールと隣接を指定できるため、詳細なレベルの設定を使用することを強く推奨します。シグナリングされないセキュア メディアの詳細なオプションを使用して、追加の帯域幅が割り当てられ、RTCP 番号のチェックは、CAC の一致条件と一致するコールだけで実行されます。グローバル オプションのように、シグナリングされないセキュア メディアは、デフォルトではディセーブルです。

Cisco IOS XE Release 2.6 では、隣接の詳細なレベルでのシグナリングされない SRTP メディアを許可するように SBC を設定する場合は、次の推奨事項に従ってください。

- 隣接がセキュア コールを許可するために信頼できる場合、**security trusted-encrypted** または **security trusted-unencrypted** コマンドを使用して、発信側と着信側がある両方の隣接を、まず SRTP パス スルー用に設定します。パススルーであるため、両側を設定する必要があります。これは SRTP コールが信頼できる隣接間で許可されるデフォルトです。
- 隣接が信頼できない場合でも、信頼できない隣接の CAC 設定で SRTP パススルーを設定することで、その隣接でシグナリングされないセキュア メディアを詳細に設定できます。**srtp support** コマンドを使用し、CAC ポリシーが適用されている隣接で SRTP コールを許可します。
- コールの両方のレッグを設定して、詳細レベルのシグナリングされないセキュア メディアをイネーブルにします。発信側では **caller secure-media** コマンドを使用し、着信側では **callee secure-media** コマンドを使用します。

（注） 状況によっては、**branch** コマンドを、**caller** または **callee** コマンドの代用として使用できます。**branch** コマンドはリリース 3.5.0 で導入されました。このコマンドの詳細については、「ダイレクト非制限 CAC ポリシーの設定」（P.139）を参照してください。

設定手順については、「シグナリングされないセキュア メディアの詳細なレベルでの設定」（P.803）を参照してください。

SRTP パススルーについて

Cisco Unified Border Element（SP Edition）は、SIP シグナリング暗号化の Transport Layer Security（TLS）および RTP メディア暗号化を提供するための Secure Real-Time Protocol（SRTP）を使用した、エンドポイント間の SIP コールをサポートしています。ただし、これら 2 つの暗号化メカニズムは同時に展開することはできず、関連設定で呼び出された必要なコール フローによって変化します。

さらに SRTP パススルー設定を掘り下げる前に、*trusted* と *untrusted* および *encrypted* と *unencrypted* の 2 つの概念について理解すると有用です。

「trusted」は、関連隣接がセキュア コールの許可について信頼されていることを意味します。標準 SIP へのコール：URI が受け入れられます。セキュア SIPS へのコール：URI が受け入れられて、信頼されている隣接でルーティングされます（暗号化または非暗号化）。「untrusted」は、関連隣接がセキュア コールの搬送について信頼されていないことを意味します。標準 SIP へのコール：URI が受け入れられます。セキュア SIPS へのコール：URI が即座に拒否されます。

「encrypted」は関連隣接が SIP シグナリングの TLS を使用していることを意味し、「unencrypted」は関連隣接が SIP シグナリングの TLS を使用していないことを意味します。

trusted/untrusted と encrypted/unencrypted は、次の 4 つの組み合わせで使用されます。これは、**security** コマンドを使用して呼び出されます。

- **untrusted-unencrypted**：隣接は信頼されず、暗号化されません。隣接は、セキュア SIP コール（SIPS URI のあるコール）を搬送するために信頼されず、SIP シグナリングの TLS 暗号を使用しません。
- **untrusted-encrypted**：隣接は信頼されず、暗号化されます。隣接は、セキュア SIP コール（SIPS URI のあるコール）を搬送するために信頼されず、SIP シグナリングの TLS 暗号を使用します。
- **trusted-unencrypted**：隣接は信頼されていて、暗号化されません。隣接は、セキュア SIP コール（SIPS URI のあるコール）を搬送するために信頼され、SIP シグナリングの TLS 暗号を使用しません。
- **trusted-encrypted**：隣接は信頼されていて、暗号化されます。隣接は、セキュア SIP コール（SIPS URI のあるコール）を搬送するために信頼され、SIP シグナリングの TLS 暗号を使用します。

Cisco Unified Border Element（SP Edition）が稼動している場合、デフォルトでは、SRTP コールが信頼されているインターフェイスをパススルーすることができます。

SRTP パススルー機能の条件は次のとおりです。

- SRTP パススルーは、両方のコール レッグに設定されている必要があります。ターゲット隣接が SRTP パススルーをサポートしていない場合、エラー メッセージ 415（未サポートのメディア タイプ）でコールが拒否されます。
- 1 つの隣接からの Invite で受信された「m= .. RTP/SAVP ..」および a="crypto:..." フィールドは、ターゲット隣接への Invite で渡されます。

- 「m= ...RTP/SAVP...」は、SBC の SRTP パススルー動作をトリガーするために Invite 内で必要なフィールドです。

次に、RFC-4568 で説明しているように、エンドポイントからの SRTP Invite および Response コールフローのサンプルを示します。

オファー側の送信：

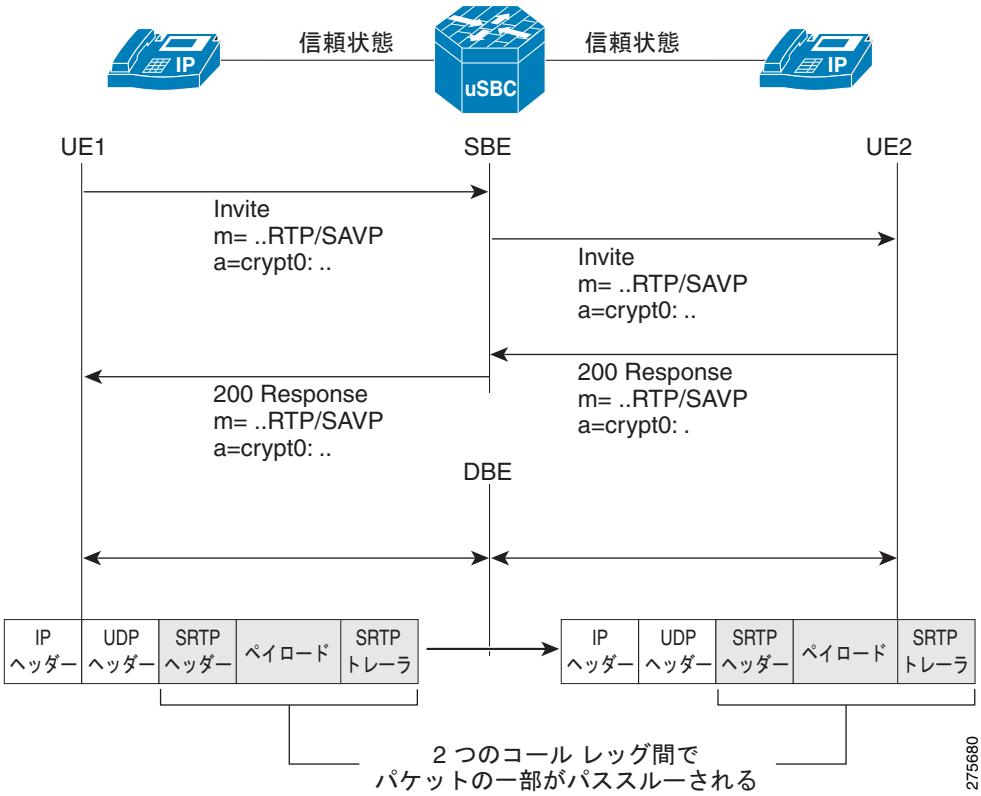
```
v=0
o=sam 2890844526 2890842807 IN IP4 10.47.16.5
s=SRTP Discussion
i=A discussion of Secure RTP
u=http://www.example.com/seminars/srtp.pdf
e=marge@example.com (Marge Simpson)
c=IN IP4 168.2.17.12
t=2873397496 2873404696
m=audio 49170 RTP/SAVP 0
a=crypto:1 AES_CM_128_HMAC_SHA1_80
 inline:WVNfX19zZW1jdGwgKCkgewkyMjA7fQp9CnVubGVz|2^20|1:4
 FEC_ORDER=FEC_SRTP
a=crypto:2 F8_128_HMAC_SHA1_80
 inline:MTIzNDU2Nzg5QUJDREUwMTIzNDU2Nzg5QUJjZGVm|2^20|1:4;
 inline:QUJjZGVmMTIzNDU2Nzg5QUJDREUwMTIzNDU2Nzg5|2^20|2:4
 FEC_ORDER=FEC_SRTP
```

アンサー側の応答：

```
v=0
o=jill 25690844 8070842634 IN IP4 10.47.16.5
s=SRTP Discussion
i=A discussion of Secure RTP
u=http://www.example.com/seminars/srtp.pdf
e=homer@example.com (Homer Simpson)
c=IN IP4 168.2.17.11
t=2873397526 2873405696
m=audio 32640 RTP/SAVP 0
a=crypto:1 AES_CM_128_HMAC_SHA1_80
 inline:PS1uQCVeeCFCanVmcjkpPywjNWhcYD0mXXtxaVBR|2^20|1:4
```

図 1 は、SRTP パス スルー コール フローを示します。

図 1　SRTP パススルー コール フロー



SRTP パススルー機能は、新しいコール アドミッション制御（CAC）エントリ変数である「srtp transport」をアドミッション コントロール テーブル内で定義します。「srtp transport」変数を設定すると、CAC ポリシーに隣接のポリシーを「allowed（許可）」、「disallowed（不許可）」、または「trust only（信頼だけ）」に設定するオプションがあります。

SRTP パススルーを使用するコールは、ポリシーによって指定された隣接上で許可されます。ポリシーが競合する場合、「disallowed」が「allowed」を無効にし、「allowed」が「trusted-only」を無効にします。CAC ポリシーを設定して「srtp transport」変数を定義しない場合、CAC ポリシーでは「trusted-only」のデフォルト値をとって、信頼されているエンドポイント間の SRTP コールを制限します。

詳細については、隣接 CAC ポリシーを設定する **srtp support** コマンドを参照してください。このコマンドの **no** 形式は、「srtp support」変数を「trusted-only」に設定します。**show sbc sbe cac-policy-set table entry** コマンドは、「SRTP Transport」フィールドを表示し、また隣接のポリシーが SRTP トランスポートに対して許可、不許可、信頼だけのいずれかであるのかを表示するように変更されます。

SRTP パススルーを許可し、次のような特定のセキュリティ ポリシングの設定を許可するように CAC ポリシーを設定できます。

- 指定隣接でのセキュア コールの回避
- 指定隣接を通じて送信されるすべてのメディアが安全であることの保証
- セキュア ストリームが安全な SIP 隣接を通じてシグナリングされていることの保証。

SRTP と RTP 間のインターワーキングおよび SRTP パススルーについて

Secure Real-Time Transport Protocol（RTP）と Real-time Transport Protocol（RTP）間のインターワーキングは、Cisco ASR 1000 シリーズ アグリゲーション サービス ルータ上のセッション ボーダー コントローラ（SBC）サービスでサポートされます。

システム管理者は、SRTP と RTP 間のインターワーキングを設定し、自身のネットワークが他のネットワークと通信できるようにし、セキュリティをネットワークに追加できます。SRTP と RTP 間のインターワーキングを使用すると、SRTP を使用するネットワークが、RTP を使用するネットワークからのコールを受け付けることができます。

SRTP と RTP 間のインターワーキング機能は、両方のタイプのネットワーク（SRTP ネットワークと RTP ネットワーク）の間でデータ ストリームを暗号化および復号化する機能を SBC に提供します。

SRTP と RTP 間のインターワーキングは、ユーザネットワーク インターフェイス（UNI）とユーザとネットワーク間インターフェイス（NNI）の両方に配置できます。

サポートされている機能

SRTP と RTP 間のインターワーキングの次の機能が SBC によってサポートされます。

- SBC で生成される SRTP 暗号化および復号化キー。
- 発信側と着信側の両方の CAC ポリシーが SRTP をサポートしている場合、SRTP パススルー、終了、再発信用の設定可能なポリシー。
- H.248 による分散 DBE モードの SRTP と RTP 間のインターワーキング。
- さまざまな SRTP プリファレンスとポリシー設定に対する SBC のコール処理を確認するための情報が格納された PD ログ。（暗号キーは PD ログに表示されません）。
- SRTP ストリームのステートフル スイッチオーバー（SSO）。

CAC ポリシーは、次の種類の RTP と SRTP 間のインターワーキングをサポートできます。

- RTP-only
- SRTP-only
- SRTP-optional
- SRTP-prefer

CAC ポリシーが SRTP-only を使用する場合

- その CAC ポリシーに関連付けられたすべてのメディア ストリームが SRTP を使用します。SRTP ストリームは、ピアの隣接が SRTP をサポートする場合、エンド ツー エンドです。ピアの隣接が SRTP をサポートしていないか、ポリシーの設定が終端と再発信に設定されている場合、SBC は必要な SRTP 暗号化および復号化を実行します。
- SBC は受信 RTP を拒否し、適切な応答コードを送信します。

CAC ポリシーが RTP-only を使用する場合

- その CAC ポリシーに関連付けられたすべてのメディア ストリームが RTP を使用します。RTP ストリームは、ピアの隣接が SRTP を義務付けていない場合、エンド ツー エンドです。ピアの隣接が SRTP を義務付けている場合、SBC は RTP と SRTP 間のインターワーキングを実行します。
- SBC は受信 SRTP を拒否し、適切な応答コードを送信します。

CAC ポリシーが SRTP-optional を使用する場合

- SRTP-optional は着信コールのネゴシエーションごとです。
- SBC は、着信 RTP および着信 SRTP のコールを受け入れます。
- 着信側 CAC ポリシーが SRTP-only を使用していない限り、RTP と RTP 間のインターワーキングは不要です。
- 着信側 CAC ポリシーが RTP-only を使用しているか、ポリシー設でパススルー モードが禁止されていない限り、着信 SRTP コールで SRTP 暗号化は不要です。

CAC ポリシーが SRTP-prefer を使用する場合

- SBC は、エンドポイントからの RTP または SRTP オファーを受け入れます。
- SBC は、エンドポイントに、受信オファーが RTP または SRTP の場合、SRTP を提供します。

次の SRTP と RTP の統計情報が収集され、グローバル レベルおよび隣接レベルの show コマンドで表示できます。

- 要求された RTP が原因で拒否されたコール数
- 要求された SRTP が原因で拒否されたコール数
- SRTP パススルーを使用するコール数
- RTP と SRTP 間のインターワーキングを実行するコール数
- RTP を使用するコール数
- SRTP を使用するコール数

SIP SRTP オファー再試行機能

srtp {branch | callee | caller} retry rtp コマンドを使用して SIP SRTP オファー再試行機能が設定されており、前の SRTP（RTP/SAVP）オファーに対する応答として 415 または 488 のリジェクト エラー コードが生成される場合、SBC は RTP（RTP/AVP）を使用してオファーを再発行します。これにより、SBC はコール レッグで SRTP を設定し、SRTP がサポートされていない場合に RTP にダウングレードを試みることができます。

（注） 415 および 488 のエラー コードは、汎用エラーです。SRTP オファー再試行機能を設定すると、SBC は 415 および 488 のエラー コードの原因が最初の RTP/SAVP オファーにあると解釈します。

SRTP オファーへのダウングレードした応答

srtp {branch | callee | caller} response downgrade コマンドを使用すると、SBC は RTP/AVP 応答を RTP/SAVP オファーに対する応答として送信し、メディアのセキュリティをダウングレードできます。たとえば、SRTP インターワーキングが CAC ポリシーで設定されておらず、発信側が RTP/SAVP を提供し、着信側が RTP/AVP で応答した場合、このコマンドにより SBC は、コールを拒否する代わりに RTP/AVP への応答ダウングレードできます。

ダウングレードを設定しない場合、SBC はオファー / アンサー プロトコルに厳密に準拠し、サポートされていない RTP/SAVP オファーを拒否します。

これは、標準でない手順であり、広くサポートされていません。SBC は、SRTP ダウングレード応答の受信を常にサポートしていますが、このダウングレード フラグが設定されている場合だけダウングレード応答を送信します。

RTP への SRTP フォールバックの場合、次の両方のケースで、全体的な側ごとの SRTP ポリシーおよび RTP-SRTP インター ワーキング ポリシーに従います。

- ポリシーで RTP がまったく許可されていない場合、SBC はフォールバックを試みません。
- ポリシーで RTP-SRTP インターワーキングが許可されていない場合、SBC は応答側でフォールバックを許可しますが、SBC が提供側でもダウングレードできる場合に限ります。

SBC による SRTP の処理方法

STRP ポリシーの振る舞いは、次のコマンドがどのように設定されているかによって異なります。

- **srtp branch forbid | mandate | allow | prefer**
- **srtp caller forbid | mandate | allow | prefer**
- **srtp callee forbid | mandate | allow | prefer**
- **srtp media interworking forbid | allow**
- **srtp interworking forbid | allow**

これらのコマンドの設定は次のように定義されます。

- **forbid**：SRTP はコールの発信側または着信側でサポートされません。
- **mandate**：SRTP はコールの発信側または着信側で必須です。
- **allow**：SRTP はコールの発信側または着信側で任意です。
- **prefer**：この隣接では SRTP が優先されます。RTP および SRTP の両方が受信で許可されますが、送信では SRTP のみが提供されます。prefer オプションがコールの提供側で設定された場合、allow と同様に機能します。prefer オプションがコールの応答側に設定され、RTP または SRTP を選択できる場合、SRTP が提供されます。

SRTP ポリシー パススルー テーブル

次の表に、コールの両側の SRTP ポリシーの設定に基づく SBC の動作を示します。

表 1 に、SRTP ポリシーがある場合に、SBC による、RTP として提供されたストリームの SRTP パススルー タイプの選択方法を示します。

表 1　SRTP ポリシーがある場合の SBC による RTP オファーの処理

SRTP ポリシー		SRTP パススルー タイプ	
提供側	応答側	不可能なインターワーキング	可能なインターワーキング
mandate	*	拒否	拒否
forbid	mandate	拒否	RTP-SRTP
forbid	forbid	RTP-RTP	RTP-RTP
forbid	allow	RTP-RTP	RTP-RTP
forbid	prefer	RTP-RTP	RTP-SRTP
allow/prefer	mandate	拒否	RTP-SRTP
allow/prefer	forbid	RTP-RTP	RTP-RTP

表 1　SRTP ポリシーがある場合の SBC による RTP オファーの処理 （続き）

SRTP ポリシー		SRTP パススルー タイプ	
提供側	応答側	不可能なインターワーキング	可能なインターワーキング
allow/prefer	allow	RTP-RTP	RTP-RTP
allow/prefer	prefer	RTP-RTP	RTP-SRTP

表 2 に、SBC による、SRTP として提供されたストリームの SRTP パススルー タイプの選択方法を示します。

表 2　SRTP オファーに対する SBC の SRTP ポリシーの処理

SRTP ポリシー		SRTP パススルー タイプ		
提供側	応答側	インターワーキングなし ダウングレードなし	可能なインターワーキング	ダウングレード可能
forbid	mandate	拒否	拒否	RTP-SRTP
forbid	forbid	拒否	拒否	RTP-RTP
forbid	allow	拒否	拒否	RTP-RTP
forbid	prefer	拒否	拒否	RTP-SRTP
mandate	mandate	SRTP-SRTP	SRTP-SRTP	SRTP-SRTP
mandate	forbid	拒否	SRTP-RTP	拒否
mandate	allow/prefer	SRTP-SRTP	SRTP-SRTP	SRTP-SRTP
allow/prefer	mandate	SRTP-SRTP	SRTP-SRTP	SRTP-SRTP
allow/prefer	forbid	拒否	SRTP-RTP	RTP-RTP（3）
allow/prefer	allow/prefer	SRTP-SRTP	SRTP-SRTP	SRTP-SRTP

表 3 に、SBC が SRTP オファーへの応答として SIP 415 または SIP 488 の拒否コードを受け取り、SRTP を RTP として再試行する機能が設定されている場合に、SBC が SRTP パス スルー タイプを選択する方法を示します。

表 3　SRTP を RTP として再試行する場合の、RTP 拒否に対する SBC の SRTP ポリシーの処理

SRTP ポリシー		SRTP パススルー タイプ	
提供側	応答側	不可能なインターワーキング	可能なインターワーキング
*	mandate	拒否	拒否
mandate	allow/prefer	SRTP-RTP	拒否
allow/prefer	allow/prefer	SRTP-RTP	RTP-RTP

表 4 に、SRTP オファーに対して RTP のダウングレード応答を受け取った場合に SBC が SRTP パススルー タイプを選択する方法を示します。

表 4　SRTP から RTP へのダウングレード応答に対する SBC の SRTP ポリシーの処理

SRTP ポリシー		SRTP パススルー タイプ	
提供側	応答側	不可能なインターワーキング	可能なインターワーキング
*	mandate	コール失敗	コール失敗
mandate	allow/prefer	SRTP-RTP	コール失敗
allow/prefer	allow/prefer	SRTP-RTP	RTP-RTP

制約事項

SRTP と RTP 間のインターワーキングおよび SRTP パススルーには次の制約事項があります。

- PacketCable イベント メッセージは、SRTP/RTP インターワーキング コールおよび SRTP パススルー コールに課金し続けますが、課金は SRTP が一方または両方のコール レッグで使用されたことを示すわけではありません。
- レイトとアーリー間のインターワーキングと SRTP と RTP 間のインターワーキングでは、SBC は生成された SDP オファーの SRTP をサポートしません。コールが RTP-RTP コールであるように強制されます。これが設定されているコール ポリシーに違反する場合、イベントが記録され、コールは設定に失敗します。
- コールに複数のストリーム（SDP に複数の m= 行）がある場合、各ストリームは別のパススルー タイプである場合があります。特定のストリームが満たされない場合、コールは拒否されます。複数のストリームと異なるパススルー タイプを持つコールは、次の場合に発生することがあります。
  - RTP および SRTP ストリームの組み合わせを含むオファーを受信した場合。
  - 応答が SRTP ストリームから RTP ストリームのサブセットにダウングレードした場合。
  - 一部のストリームがインターワーキングを必要とし、他のストリームは必要としない場合。
- SRTP 機能は H.248 でシグナリングされないため、SBC によって自動的に検出できません。この機能は、SBC に手動で設定する必要があります。
- SBC による MG の選択では、MG がサポートするクリプト スイートに基づく MG の選択は行われません。
- SBC では、ユーザがコールごとに個別の SRTP セッション パラメータを設定することはできません。
- SBC の SRTP 機能は、シグナリングされない SRTP と連動しません。
- SBC では、SIP フォーキング応答を受信すると SRTP コールが失敗します。
- レイトとアーリー間のインターワーキングは SRTP をサポートしません。
- H.323 と SIP 間のインターワーキングは SRTP をサポートしません。
- SBC は、RTP-SRTP コールで、RFC5027 セキュリティ プレコンディション シグナリングを終端できません。
- SBC は SRTP コールの市内電話転送をサポートしていません。
- SBC は現在 AES_CM_128_HMAC_SHA1_32 クリプト スイートのみをサポートします。
- SBC は自身が生成するマスター キーを更新しません。
- SBC はパケット使用数に達するとマスター キー ローテーションを再ネゴシエートしません（RFC3711 で規定）。

- トランス コーダが SRTP をサポートしない場合（MGX など）、SBC は SRTP-SRTP コールを許可しません。SBC は、トランスコーダのいずれの側でも 2 つのメディア ゲート上の SRTP-RTP インターワーキングを実行できません。
- RTP-SRTP および SRTP-RTP コールはサードパーティのトランスコーダでトランスコードできます。その場合、トランスコーダ RTP を介するメディアとインターワーキングは、SRTP エンドポイントに最も近い側の SBC で行われます。

RTP と SRTP 間のインターワーキングを設定するには、「SRTP と RTP 間のインターワーキングのための CAC ポリシーの設定」（P.813）および「SRTP と RTP 間のインターワーキングのための CAC ポリシーの設定例」（P.823）を参照してください。

SRTP 用に更新されたこの既存のコマンドを使用して、指定した送信元隣接のポリシー障害の統計情報を表示できます。

show sbc sbe call-stats src-adjacency

SRTP 用に更新されたこの既存のコマンドを使用して、SBE 上のすべてのコールを表示できます。

show sbc sbe calls srtp-iw

グローバル レベルでのセキュア メディアの設定

（注） CAC テーブル エントリ コマンドを使用して詳細なレベルでセキュア メディアを設定できるように、シグナリングされないセキュア メディア機能が Cisco IOS XE Release 2.6 に導入されました。この機能の導入により、グローバル レベルのセキュア メディアの設定機能が廃止されました。リリース 2.6 よりも前のリリースからアップグレードする場合は、「シグナリングされないセキュア メディアの詳細なレベルでの設定」（P.803）に記載されている手順を参照してください。

セキュア メディアをグローバルに設定するには、次の手順を実行します。

手順の概要

1. **configure terminal**
2. **sbc** *sbc-name*
3. **sbe**
4. **secure-media**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** **例：** `Router# configure terminal`	グローバル コンフィギュレーション モードをイネーブルにします。
ステップ 2	**sbc** *sbc-name* **例：** `Router(config)# sbc mysbc`	Cisco Unified Border Element（SP Edition）で SBC サービスを作成し、SBC コンフィギュレーション モードを開始します。
ステップ 3	**sbe** **例：** `Router(config-sbc)# sbe`	SBC の Signaling Border Element（SBE）機能のモードを開始します。

ステップ 4	`secure-media`
	
	例：
	`Router(config-sbc-sbe)# secure-media`
		すべてのメディア フローを暗号化メディア フローとして扱うように SBC を設定します。これにより、DTLS や SRTP パケットなどのメディア パケットで SBC をパススルーすることができるようになります。

ステップ 5	**end** **例：** `Router(config-sbc-sbe)# end`	SBE モードを終了し、特権 EXEC モードに戻ります。

シグナリングされないセキュア メディアの詳細なレベルでの設定

シグナリングされないセキュア メディアをイネーブルにするように設定された CAC ポリシーを使用し、両方の隣接および両方のコール レッグを詳細なレベルで設定するには、次の手順を実行します。

（注） この手順では、**caller** コマンドと **callee** コマンドが使用されています。シナリオによっては、**caller** と **callee** のコマンド ペアの代わりに **branch** コマンドを使用できます。**branch** コマンドはリリース 3.5.0 で導入されました。このコマンドの詳細については、「ダイレクト非制限 CAC ポリシーの設定」（P.139）を参照してください。

手順の概要

1. **configure terminal**
2. **sbc** *sbc-name*
3. **sbe**
4. **adjacency {sip | h323}** *adjacency-name*
5. **security** [**untrusted** | **trusted-encrypted** | **untrusted-encrypted** | **trusted-unencrypted**]
6. **exit**

7. **adjacency {sip | h323}** *adjacency-name*
8. **security** [**untrusted** | **trusted-encrypted** | **untrusted-encrypted** | **trusted-unencrypted**]
9. **exit**
10. **cac-policy-set** *policy-set-id*
11. **first-cac-table** *table-name*
12. **cac-table** *table-name*
13. **table-type limit** *list of limit tables*
14. **entry** *entry-id*
15. **match-value** *key*
16. **srtp support [allow | disallow | trusted-only**]
17. **caller secure-media**
18. **callee secure-media**
19. **action** {**cac-complete | next-table** *goto-table-name*}
20. **exit**
21. **complete**
22. **exit**
23. **active-cac-policy-set** *policy-set-id*
24. **end**
25. **show sbc** *sbc-name* **sbe cac-policy-set** [id [table name [entry id]] | active [table name [entry id]]] [detail]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** **例：** `Router# configure terminal`	グローバル コンフィギュレーション モードをイネーブルにします。
ステップ 2	**sbc** *sbc-name* **例：** `Router(config)# sbc mysbc`	Cisco Unified Border Element（SP Edition）で SBC サービスを作成し、SBC コンフィギュレーション モードを開始します。
ステップ 3	**sbe** **例：** `Router(config-sbc)# sbe`	SBC の Signaling Border Element（SBE）機能のモードを開始します。
ステップ 4	**adjacency** {**sip** | **h323**} *adjacency-name* **例：** `Router(config-sbc-sbe)# adjacency sip client`	この例では、発信側の SIP 隣接（名前は「client」）を設定します。また、SBE SIP 隣接のモード（別名隣接 SIP モード）を開始します。

	コマンドまたはアクション	目的
ステップ 5	**security [untrusted | trusted-encrypted | untrusted-encrypted | trusted-unencrypted]** **例：** `Router(config-sbc-sbe-adj-sip)# security trusted-encrypted`	SIP 隣接でトランスポートレベルのセキュリティ（TLS）を設定します。 詳細レベルのセキュア メディアでは、信頼できる隣接を trusted-encrypted または trusted-unencrypted として設定します。 trusted は、隣接がセキュアな SIP コール（SIPS URI を使用したコール）を搬送することが信頼されることを意味します。encrypted は、隣接が SIP シグナリングに TLS 暗号化を使用することを意味します。unencrypted は、SIP シグナリングで TLS 暗号化を使用しないことを意味します。 **（注）** この隣接が *untrusted* の場合、ステップ 〜 を省略します。CAC ポリシー テーブルで信頼できない隣接を設定する必要があります。

ステップ 6	`exit`
	
	例：
	`Router(config-sbc-sbe-adj-sip)# exit`
		SBE SIP 隣接モードを終了して、SBE モードに戻ります。

ステップ 7	**adjacency** {**sip** | **h323**} *adjacency-name* **例：** `Router(config-sbc-sbe)# adjacency sip server`	この例では、着信側の SIP 隣接（名前は「server」）を設定します。また、SBE SIP 隣接のモード（別名隣接 SIP モード）を開始します。
ステップ 8	**security** [**untrusted** | **trusted-encrypted** | **untrusted-encrypted** | **trusted-unencrypted**] **例：** `Router(config-sbc-sbe-adj-sip)# security trusted-unencrypted`	SIP 隣接でトランスポートレベルのセキュリティ（TLS）を設定します。 詳細レベルのセキュア メディアでは、信頼できる隣接を trusted-encrypted または trusted-unencrypted として設定します。 trusted は、隣接がセキュアな SIP コール（SIPS URI を使用したコール）を搬送することが信頼されることを意味します。encrypted は、隣接が SIP シグナリングに TLS 暗号化を使用することを意味します。unencrypted は、SIP シグナリングで TLS 暗号化を使用しないことを意味します。 **（注）** この隣接が *untrusted* の場合、ステップ 〜 を省略します。CAC ポリシー テーブルで信頼できない隣接を設定する必要があります。
ステップ 9	**exit** **例：** `Router(config-sbc-sbe-adj-sip)# exit`	SBE SIP 隣接モードを終了して、SBE モードに戻ります。
ステップ 10	**cac-policy-set** *policy-set-id* **例：** `Router(config-sbc-sbe)# cac-policy-set 1`	SBE エンティティ内で CAC ポリシー セット コンフィギュレーション モードを開始して、必要に応じて新規ポリシー セットを作成します。 *policy-set-id*：ポリシー セットを特定するためにユーザによって選択される整数。範囲は 1 〜 2147483647 です。

	コマンドまたはアクション	目的
ステップ 11	**first-cac-table** *table-name* **例：** `Router(config-sbc-sbe-cacpolicy)# first-cac-table testSecure`	処理する最初のポリシー テーブルの名前を設定します。CAC ポリシーには、設定済みのテーブルが数多くあります。CAC ポリシーのアプリケーションを開始するには、使用される最初のテーブルを定義する必要があります。 *table-name*：最初に処理すべきアドミッション コントロール テーブル。
ステップ 12	**cac-table** *table-name* **例：** `Router(config-sbc-sbe-cacpolicy)# cac-table testSecure`	SBE ポリシー セットのコンテキスト内で、アドミッション コントロール テーブル（必要に応じて作成します）のコンフィギュレーション モードを開始します。 *table-name*：アドミッション コントロール テーブル名。
ステップ 13	**table-type limit** ***list of limit tables*** **例：** `Router(config-sbc-sbe-cacpolicy-cactable)# table-type limit all`	エントリを一致させるのに使用される基準を入力する新規 CAC 制限テーブルを設定します。 *list of limit tables* には、次のいずれかの値を指定できます。 • **account**：アカウント名を比較します。 • **adj-group**：隣接グループ名を比較します。 • **adjacency**：隣接名を比較します。 • **all**：比較タイプはありません。すべてのイベントがこのタイプと一致します。 • **call-priority**：コール プライオリティと比較します。 • **category**：番号分析が割り当てられたカテゴリを比較します。 • **dst-account**：宛先アカウント名を比較します。 • **dst-adj-group**：宛先隣接グループ名を比較します。 • **dst-adjacency**：宛先隣接名を比較します。 • **dst-prefix**：着信ディジット ストリングの先頭を比較します。 • **event-type**：CAC ポリシー イベント タイプと比較します。 • **src-account**：送信元アカウント名を比較します。 • **src-adj-group**：送信元隣接グループ名を比較します。 • **src-adjacency**：送信元隣接名を比較します。 • **src-prefix**：発番号ストリングの先頭を比較します。
ステップ 14	**entry** *entry-id* **例：** `Router(config-sbc-sbe-cacpolicy-cactable)# entry 1`	アドミッション コントロール テーブル内のエントリを変更するモードを開始します。 *entry-id*：テーブル エントリを指定します。

	コマンドまたはアクション	目的

ステップ 15	**match-value** *key*
	
	例：
	`Router(config-sbc-sbe-cacpolicy-cactable-entry)`
	`# match-value call-update`
		CAC 制限テーブル タイプにあるエントリの照合値を設定します。

ステップ 16	**srtp support [allow | disallow | trusted-only]**
	
	例：
	`Router(config-sbc-sbe-cacpolicy-cactable-entry)`
	`# srtp support allow`
		隣接が *untrusted* であり、細かいレベルでセキュア メディアを使用する場合、この手順を設定する必要があります。**srtp support allow** を使用して設定すると、CAC ポリシーを適用する信頼できない隣接で SRTP コールが許可されます。に進みます。
		
		CAC テーブル内の srtp support 変数を設定して、ポリシーが適用されている隣接におけるセキュア メディアの SRTP パススルーを許可または不許可にします。
		• **allow**：イベントがこの CAC ポリシーと一致する場合 SRTP トランスポートを許可します。
		• **disallow**：イベントがこの CAC ポリシーと一致する場合 SRTP トランスポートを許可しません。
		• **trusted-only**：イベントがこの CAC ポリシーと一致する場合、信頼される隣接（デフォルト）の SRTP トランスポートを許可します。
		
		SRTP パススルーを使用するコールは、ポリシーによって指定された隣接上で許可されます。

ステップ 17	**caller secure-media**
	
	例：
	`Router(config-sbc-sbe-cacpolicy-cactable-entry)`
	`# caller secure-media`
		発信側のセキュア メディア コールを設定します。

ステップ 18	**callee secure-media**
	
	例：
	`Router(config-sbc-sbe-cacpolicy-cactable-entry)`
	`# callee secure-media`
		着信側のセキュア メディア コールを設定します。

ステップ 19	**action {cac-complete | next-table** ***goto-table-name*****}**
	
	例：
	`Router(config-sbc-sbe-cacpolicy-cactable-entry)`
	`# action cac-complete`
		アドミッション コントロール テーブルのこのエントリの後で実行するアクションを設定します。各エントリは、一致基準とアクションが必要です。アクションは、トランスポートを受け入れるためのものです。
		
		アクションは次のいずれかです。
		• **cac-complete**：イベントが一致すると、この CAC ポリシーが完了します。
		• **next-table**：次の cac テーブル名を指定します。
		• *goto-table-name*：処理する次の CAC テーブルを識別するテーブル名を指定します（または処理が停止している場合は cac-complete）。

	コマンドまたはアクション	目的
ステップ 20	**exit** **例：** `Router(config-sbc-sbe-cacpolicy-cactable-entry)# exit`	CAC テーブル エントリ モードを終了し、CAC ポリシー セット コンフィギュレーション モードを開始します。
ステップ 21	**complete** **例：** `Router(config-sbc-sbe-cacpolicy)# complete`	フル セットの確定後、CAC-policy セットを終了します。
ステップ 22	**exit** **例：** `Router(config-sbc-sbe-cacpolicy)# exit`	CAC Policy-set コンフィギュレーション モードを終了し、SBE モードを開始します。
ステップ 23	**active-cac-policy-set** *policy-set-id* **例：** `Router(config-sbc-sbe)# active-cac-policy-set 1`	新規作成された CAC ポリシーをアクティブに設定します。ポリシーがアクティブの場合、変更することはできなくなります。 *policy-set-id* : アクティブにするポリシー セットを識別します。範囲は 1 ～ 2147483647 です。
ステップ 24	**end** **例：** `Router(config-sbc-sbe)# end`	SBE モードを終了し、特権 EXEC モードに戻ります。
ステップ 25	**show sbc** *name* **sbe cac-policy-set** [*id* [**table** *name* [**entry** *id*]]|**active** [**table** *name* [**entry** *id*]]] [**detail**] **例：** `Router# show sbc mysbc sbe cac-policy-set 1 detail`	CAC ポリシー テーブルの特定エントリの詳細な情報を表示します。この例では、発信側と着信側のシグナリングされないセキュア メディアが含まれます。セキュア メディア コールの両方の隣接に対するセキュリティ trusted-unencrypted が許可されます。

SRTP パススルーの設定

次の手順は、SRTP パススルーを許可するように CAC ポリシーを設定する方法を示します。

手順の概要

1. **configure terminal**
2. **sbc** *sbc-name*
3. **sbe**
4. **cac-policy-set** ***policy-set-id***
5. **first-cac-scope** ***scope-name***
6. **first-cac-table** ***table-name***
7. **cac-table** ***table-name***
8. **table-type limit** ***list of limit tables***

9. **entry** *entry-id*
10. **match-value** *key*
11. **srtp support [allow | disallow | trusted-only]**
12. **action [cac-complete | next-table |** *goto-table-name* **]**
13. **exit**
14. **exit**
15. **complete**
16. **exit**
17. **active-cac-policy-set** *policy-set-id*
18. **end**
19. **show sbc** *sbc-name* **sbe cac-policy-set** *id* **table** *name* **entry** *entry*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** **例：** `Router# configure terminal`	グローバル コンフィギュレーション モードをイネーブルにします。
ステップ 2	**sbc** *sbc-name* **例：** `Router(config)# sbc mysbc`	Cisco Unified Border Element（SP Edition）で SBC サービスを作成し、SBC コンフィギュレーション モードを開始します。
ステップ 3	**sbe** **例：** `Router(config-sbc)# sbe`	SBC の Signaling Border Element（SBE）機能のモードを開始します。
ステップ 4	**cac-policy-set** *policy-set-id* **例：** `Router(config-sbc-sbe)# cac-policy-set 1`	SBE エンティティ内で CAC ポリシー セット コンフィギュレーション モードを開始して、必要に応じて新規ポリシー セットを作成します。 *policy-set-id*：ポリシー セットを特定するためにユーザによって選択される整数。範囲は 1 ～ 2147483647 です。

	コマンドまたはアクション	目的
ステップ 5	**first-cac-scope** *scope-name* **例：** `Router(config-sbc-sbe-cacpolicy)# first-cac-scope call`	ポリシーのアドミッション コントロール ステージの実行時に制限が最初に定義されるべき範囲を設定します。各 CAC ポリシーには、これに適用される範囲があります。この CAC ポリシーは、コール ベース単位で適用されます。 *scope-name* には、次のいずれかの値が指定されます。 • **adj-group**：同じ隣接グループのメンバーからのイベントの制限。 • **call**：制限が単一コール単位です。 • category：カテゴリ単位の制限。 • **dst-account**：同じアカウントに送信されるイベントの制限。 • **dst-adj-group**：同じ隣接グループに送信されるイベントの制限。 • **dst-adjacency**：同じ隣接に送信されるイベントの制限。 • **dst-number**：同一隣接グループ番号を持つイベントの制限。 • **global**：制限がグローバルです（他のオプションと組み合わせることができません）。 • **src-account**：同じアカウントからのイベントの制限。 • **src-adj-group**：同じ隣接グループからのイベントの制限。 • **arc-adjacency**：同じ隣接からのイベントの制限。 • **src-number**：同じ送信元番号を持つイベントの制限。
ステップ 6	**first-cac-table** *table-name* **例：** `Router(config-sbc-sbe-cacpolicy)# first-cac-table testSecure`	処理する最初のポリシー テーブルの名前を設定します。CAC ポリシーには、設定済みのテーブルが数多くあります。CAC ポリシーのアプリケーションを開始するには、使用される最初のテーブルを定義する必要があります。 *table-name*：最初に処理すべきアドミッション コントロール テーブル。
ステップ 7	**cac-table** *table-name* **例：** `Router(config-sbc-sbe-cacpolicy)# cac-table testSecure`	SBE ポリシー セットのコンテキスト内で、アドミッション コントロール テーブル（必要に応じて作成します）のコンフィギュレーション モードを開始します。 *table-name*：アドミッション コントロール テーブル名。

	コマンドまたはアクション	目的
ステップ 8	**table-type limit** *list of limit tables* **例：** `Router(config-sbc-sbe-cacpolicy-cactable)# table-type limit all`	エントリを一致させるのに使用される基準を入力する新規 CAC 制限テーブルを設定します。 *list of limit tables* には、次のいずれかの値を指定できます。 • **account**：アカウント名を比較します。 • **adj-group**：隣接グループ名を比較します。 • **adjacency**：隣接名を比較します。 • **all**：比較タイプはありません。すべてのイベントがこのタイプと一致します。 • **call-priority**：コール プライオリティと比較します。 • **category**：番号分析が割り当てられたカテゴリを比較します。 • **dst-account**：宛先アカウント名を比較します。 • **dst-adj-group**：宛先隣接グループ名を比較します。 • **dst-adjacency**：宛先隣接名を比較します。 • **dst-prefix**：着信ディジット ストリングの先頭を比較します。 • **event-type**：CAC ポリシー イベント タイプと比較します。 • **src-account**：送信元アカウント名を比較します。 • **src-adj-group**：送信元隣接グループ名を比較します。 • **src-adjacency**：送信元隣接名を比較します。 • **src-prefix**：発番号ストリングの先頭を比較します。
ステップ 9	**entry** *entry-id* **例：** `Router(config-sbc-sbe-cacpolicy-cactable)# entry 1`	アドミッション コントロール テーブル内のエントリを変更するモードを開始します。 *entry-id*：テーブル エントリを指定します。
ステップ 10	**match-value** *key* **例：** `Router(config-sbc-sbe-cacpolicy-cactable-entry)# match-value call-update`	CAC 制限テーブル タイプにあるエントリの照合値を設定します。

	コマンドまたはアクション	目的
ステップ 11	**srtp support [allow | disallow | trusted-only]** **例：** `Router(config-sbc-sbe-cacpolicy-cactable-entry)# srtp support allow`	CAC テーブル内の srtp support 変数を設定して、ポリシーが適用されている隣接におけるセキュア メディアの SRTP パススルーを許可または不許可にします。 • **allow**：イベントがこの CAC ポリシーと一致する場合 SRTP トランスポートを許可します。 • **disallow**：イベントがこの CAC ポリシーと一致する場合 SRTP トランスポートを許可しません。 • **trusted-only**：イベントがこの CAC ポリシーと一致する場合、信頼される隣接（デフォルト）の SRTP トランスポートを許可します。 SRTP パススルーを使用するコールは、ポリシーによって指定された隣接上で許可されます。ポリシーが競合する場合、「disallowed」が「allowed」を無効にし、「allowed」が「trusted-only」を無効にします。
ステップ 12	**action** [**cac-complete** | **next-table** *goto-table-name*] **例：** `Router(config-sbc-sbe-cacpolicy-cactable-entry)# action cac-complete`	アドミッション コントロール テーブルのこのエントリの後で実行するアクションを設定します。各エントリは、一致基準とアクションが必要です。アクションは、トランスポートを受け入れるためのものです。 アクションは次のいずれかです。 • **cac-complete**：イベントが一致すると、この CAC ポリシーが完了します。 • **next-table**：次の cac テーブル名を指定します。 • *goto-table-name*：処理する次の CAC テーブルを識別するテーブル名を指定します（または処理が停止している場合は cac-complete）。
ステップ 13	**exit** **例：** `Router(config-sbc-sbe-cacpolicy-cactable-entry)# exit`	CAC テーブル エントリ サブモードを終了して、cacpolicy cactable モードを開始します。
ステップ 14	**exit** **例：** `Router(config-sbc-sbe-cacpolicy-cactable)# exit`	cacpolicy cactable サブモードを終了して、cacpolicy モードを開始します。
ステップ 15	**complete** **例：** `Router(config-sbc-sbe-cacpolicy)# complete`	CAC テーブル内のすべてのエントリが設定された後に CAC ポリシーを完了します。
ステップ 16	**exit** **例：** `Router(config-sbc-sbe-cacpolicy)# exit`	cacpolicy サブモードを終了して、SBE モードを開始します。

	コマンドまたはアクション	目的
ステップ 17	**active-cac-policy-set** *policy-set-id* **例：** `Router(config-sbc-sbe)# active-cac-policy-set 1`	新規作成された CAC ポリシーをアクティブに設定します。ポリシーがアクティブの場合、変更することはできなくなります。 *policy-set-id*：アクティブにするポリシー セットを識別します。範囲は 1 ～ 2147483647 です。
ステップ 18	**end** **例：** `Router(config-sbc-sbe)# end`	SBE モードを終了し、特権 EXEC モードに戻ります。
ステップ 19	`show sbc` *sbc-name* `sbe cac-policy-set` *id* `table` *name* `entry` *entry* **例：** `Router# show sbc mysbc sbe cac-policy-set 1 table testSecure entry 1`	「SRTP Transport」フィールドや、SRTP トランスポートに対して隣接のポリシーが許可されているか、不許可か、または信頼だけなのかを含む、詳細出力を表示します。

SRTP と RTP 間のインターワーキングのための CAC ポリシーの設定

RTP と SRTP 間のインターワーキング用にコールの発信側と着信側の CAC ポリシーを設定するには、次の手順を実行します。

手順の概要

1. **configure terminal**
2. **sbc** *sbc-name*
3. **sbe**
4. **cac-policy-set** *policy-set-id*
5. **first-cac-table** *table-name*

コールの発信側の CAC テーブル

6. **cac-table** *table-name*
7. **table-type limit** *list of limit tables*

(必要に応じてステップ 8 ～ 14 を繰り返します)

8. **entry** *entry-id*
9. **match-value** *key*
10. **srtp support allow**
11. **action next-table** *goto-table-name*
12. **srtp caller forbid | mandate | allow | prefer**
13. **srtp interworking forbid | allow**
14. **srtp media interworking forbid | allow**

コールの着信側の CAC テーブル

15. **cac-table** *table-name*
16. **table-type limit** *list of limit tables*

（必要に応じてステップ 17 ～ 23 を繰り返します）

17. **entry** *entry-id*
18. **match-value** *key*
19. **srtp support allow**
20. **action cac-complete**
21. **srtp callee forbid | mandate | allow**
22. **srtp interworking forbid | allow**
23. **srtp media interworking forbid | allow**

*（***complete** *コマンドは、すべてのエントリを設定した後に実行します）*

24. **complete**
25. **end**
26. **show sbc** *name* **sbe cac-policy-set** *id* **detail**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** **例：** `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**sbc** *sbc-name* **例：** `Router(config)# sbc SBC1`	Cisco Unified Border Element（SP Edition）で SBC サービスを作成し、SBC コンフィギュレーション モードを開始します。
ステップ 3	**sbe** **例：** `Router(config-sbc)# sbe`	SBC の Signaling Border Element（SBE）機能のモードを開始します。
ステップ 4	**cac-policy-set** *policy-set-id* **例：** `Router(config-sbc-sbe)# cac-policy-set 44`	SBE エンティティ内で CAC ポリシー セット コンフィギュレーション モードを開始して、新規ポリシー セットを作成します。 • *policy-set-id*：ポリシー セットを特定するためにユーザによって選択される整数。指定できる範囲は 1 ～ 2147483647 です。
ステップ 5	**first-cac-table** *table-name* **例：** `Router(config-sbc-sbe-cacpolicy)# first-cac-table 44`	どの CAC テーブルを最初に処理するかを指定します。 • *table-name*：最初に処理するテーブル名。

コマンドまたはアクション	目的
コールの発信側の CAC テーブル	
ステップ 6 **cac-table** *table-name* **例：** `Router(config-sbc-sbe-cacpolicy)# cac-table 44`	SBE ポリシー セットのコンテキスト内で、アドミッション コントロール テーブル（必要に応じて作成します）のコンフィギュレーション モードを開始します。 • *table-name*：アドミッション コントロール テーブル名。
ステップ 7 **table-type limit** *list of limit tables* **例：** `Router(config-sbc-sbe-cacpolicy-cactable)# table-type limit src-adjacency`	**match-value** コマンドで照合するテーブル タイプの制限を設定します。この例では、次のテーブル タイプを使用します。 • *src-adjacency*：送信元隣接名を比較します。

必要なエントリを設定するのに必要な回数だけ、ステップ 8 ～ 14 を繰り返します。

ステップ 8 **entry** *entry-id* **例：** `Router(config-sbc-sbe-cacpolicy-cactable)# entry 1`	アドミッション コントロール テーブル内のエントリを変更するモードを開始します。 • *entry-id*：テーブル エントリを指定します。
ステップ 9 **match-value** *key* **例：** `Router(config-sbc-sbe-cacpolicy-cactable-entry)# match-value A`	コール アドミッション制御（CAC）制限テーブルにあるエントリの照合値を設定します。 • *key*：イベントの照合に使用するキーワードを指定します。キーの形式は、table-type 制限によって決定されます。
ステップ 10 **srtp support allow** **例：** `Router(config-sbc-sbe-cacpolicy-cactable-entry)# srtp support allow`	SRTP サポートを設定します。
ステップ 11 **action next-table** *goto-table-name* **例：** `Router(config-sbc-sbe-cacpolicy-cactable-entry)# action next-table 45`	このルーティング エントリが選択された場合に実行するアクションを設定します。 • *goto-table-name*：イベントがエントリに一致する場合に処理する次のルーティング テーブルを指定します。
ステップ 12 **srtp caller forbid | mandate | allow | prefer** **例：** `Router(config-sbc-sbe-cacpolicy-cactable-entry)# srtp caller forbid`	次のいずれかの SRTP 設定を使用して、コールの発信側の SRTP を設定します。 • **forbid**：SRTP はコールの発信側でサポートされません。 • **mandate**：SRTP はコールの発信側で必須です。 • **allow**：SRTP はコールの発信側で任意です。 • **prefer**：この隣接では SRTP が優先されます。RTP および SRTP の両方が受信で許可されますが、送信では SRTP のみが提供されます。

	コマンドまたはアクション	目的
ステップ 13	**srtp interworking forbid | allow** **例：** `Router(config-sbc-sbe-cacpolicy-cactable-entry) # srtp interworking allow`	SRTP と RTP 間のインターワーキングを設定します。 • **forbid**：コールに対し SRTP と RTP 間のインターワーキングを禁止します。 • **allow**：コールに対し SRTP と RTP 間のインターワーキングを許可します。
ステップ 14	**srtp media interworking forbid | allow** **例：** `Router(config-sbc-sbe-cacpolicy-cactable-entry) # srtp media interworking allow`	SRTP と RTP 間のメディア インターワーキングを設定します。 • **forbid**：コールに対し SRTP と RTP 間のメディア インターワーキングを禁止します。 • **allow**：コールに対し SRTP と RTP 間のメディア インターワーキングを許可します。

コールの着信側の CAC テーブル

ステップ 15	**cac-table** *table-name* **例：** `Router(config-sbc-sbe-cacpolicy)# cac-table 45`	SBE ポリシー セットのコンテキスト内で、アドミッション コントロール テーブル（必要に応じて作成します）のコンフィギュレーション モードを開始します。 • *table-name*：アドミッション コントロール テーブル名。
ステップ 16	**table-type limit** *list of limit tables* **例：** `Router(config-sbc-sbe-cacpolicy-cactable)# table-type limit src-adjacency`	**match-value** コマンドで照合するテーブル タイプの制限を設定します。この例では、次のテーブル タイプを使用します。 • *src-adjacency*：送信元隣接名を比較します。

必要なエントリを設定するのに必要な回数だけ、ステップ 17 ～ 23 を繰り返します。

ステップ 17	**entry** *entry-id* **例：** `Router(config-sbc-sbe-cacpolicy-cactable)# entry 1`	アドミッション コントロール テーブル内のエントリを変更するモードを開始します。 • *entry-id*：テーブル エントリを指定します。
ステップ 18	**match-value** *key* **例：** `Router(config-sbc-sbe-cacpolicy-cactable-entry) # match-value A`	コール アドミッション制御（CAC）制限テーブルにあるエントリの照合値を設定します。 • *key*：イベントの照合に使用するキーワードを指定します。キーの形式は、table-type 制限によって決定されます。
ステップ 19	**srtp support allow** **例：** `Router(config-sbc-sbe-cacpolicy-cactable-entry) # srtp support allow`	SRTP サポートを設定します。

	コマンドまたはアクション	目的
ステップ 20	**action next-table** *goto-table-name* **例：** `Router(config-sbc-sbe-cacpolicy-cactable-entry)# action next-table 45`	このルーティング エントリが選択された場合に実行するアクションを設定します。 • *goto-table-name*：イベントがエントリに一致する場合に処理する次のルーティング テーブルを指定します。
ステップ 21	**srtp callee forbid | mandate | allow | prefer** **例：** `Router(config-sbc-sbe-cacpolicy-cactable-entry)# srtp callee forbid`	コールの着信側の SRTP を設定します。 • **forbid**：SRTP はコールの着信側でサポートされません。 • **mandate**：SRTP はコールの着信側で必須です。 • **allow**：SRTP はコールの着信側で任意です。 • **prefer**：この隣接では SRTP が優先されます。RTP および SRTP の両方が受信で許可されますが、送信では SRTP のみが提供されます。
ステップ 22	**srtp interworking forbid | allow** **例：** `Router(config-sbc-sbe-cacpolicy-cactable-entry)# srtp interworking allow`	SRTP と RTP 間のインターワーキングを設定します。 • **forbid**：コールに対し SRTP と RTP 間のインターワーキングを禁止します。 • **allow**：コールに対し SRTP と RTP 間のインターワーキングを許可します。
ステップ 23	**srtp media interworking forbid | allow** **例：** `Router(config-sbc-sbe-cacpolicy-cactable-entry)# srtp media interworking allow`	SRTP と RTP 間のメディア インターワーキングを設定します。 • **forbid**：コールに対し SRTP と RTP 間のメディア インターワーキングを禁止します。 • **allow**：コールに対し SRTP と RTP 間のメディア インターワーキングを許可します。

***complete** コマンドは、すべてのエントリを設定した後にのみ実行します。*

ステップ 24	`complete`
	
	例：
	`Router(config-sbc-sbe-cacpolicy-cactable-entry)# complete`
		すべてのエントリを入力した後に CAC-policy を完了します。

	コマンドまたはアクション	目的
ステップ 25	**end** **例：** `Router(config-sbc-sbe-cacpolicy-cactable-entry)# end`	コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

ステップ 26	**show sbc** *name* **sbe cac-policy-set** *id* **detail**
	
	例：
	`Router# show sbc SBC1 sbe cac-policy-set 1 detail`
		CAC ポリシー テーブルの特定エントリ ID の詳細な情報を表示します。この例では、SRTP-RTP インターワーキングのデフォルト値を表示します。次に例を示します。
		`Caller SRTP support: Inherit (default)`
		`Callee SRTP support: Inherit (default)`
		`SRTP Interworking: Inherit (default)`
		`SRTP media Interworking: Inherit (default)`

RTP で多重化された RTCP の SRTP サポート

以前のリリースでは、SBC はそれぞれ異なる UDP チャネル上で送信された受信 RTP および RTCP ストリームを処理できました。リリース 3.4S から、SBC は、RTP ストリームに多重化され 1 つの UDP チャネルを介して送信された RTCP ストリームも処理できます。SBC は、RTCP と RTP のストリームを、各ストリームのペイロード形式を調べることで区別します。これは、SRTP ストリームに多重化された SRTCP ストリームにも適用されます。

（注） RFC 5761 は、RTP ストリームに多重化された RTCP ストリームについて規定しています。同じ原則が SRTCP および SRTP に適用されます。

この機能は、SBC を通してリンクされた RTP ベースおよび SRTP ベースのエンド ポイントのインターワーキングのサポートの拡張です。Cisco TelePresence System は RTP ベースのエンドポイントの例で、Cisco Umi TelePresence は SRTP ベースのエンドポイントの例です。この機能の導入により、SBC は Cisco TelePresence System から受信する RTP ストリームに多重化された RTCP ストリームを処理します。同様に、SBC は、Cisco Umi TelePresence からの SRTP ストリームに多重化された SRTCP ストリームを識別し、正常に処理します。

デフォルトでは、RTP ストリームに多重化された RTCP ストリームの検出は SBC でディセーブルになっています。次の項で説明する手順を実行して、この機能をイネーブルにできます。

RTP に多重化された RTCP の検出の設定

この作業は、RTP ストリームに多重化された RTCP ストリーム検出の設定方法について説明します。

（注） 同じ手順を、SRTP ストリームに多重化された SRTCP ストリームの検出を設定するために使用できます。

手順の概要

1. **configure terminal**
2. **sbc** *sbc-name*
3. **sbe**
4. **rtcp-mux**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** **例：** `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**sbc** *sbc-name* **例：** `Router(config)# sbc MySbc`	SBC サービス モードを開始します。 • *sbc-name*：SBC の名前。
ステップ 3	**sbe** **例：** `Router(config-sbc)# sbe`	SBE コンフィギュレーション モードを開始します。
ステップ 4	**rtcp-mux** **例：** `Router(config-sbc-sbe)# rtcp-mux`	RTP ストリームに多重化された RTCP ストリームの検出をイネーブルにします。 デフォルトでは、この機能はディセーブルになっています。

SRTP による SSRC ベースの多重化のサポート

Cisco TelePresence System などの SBC エンドポイントは、1 つの UDP チャネルに同じタイプ（音声またはビデオ）の RTP ストリームを多重化します。単一の送信元から送信された個別の RTP ストリームを区別するために、RTP ストリームの 32 ビット同期ソース（SSRC）フィールドを使用します。

SRTP ベースまたは RTP ベースのエンドポイントが単一の UDP チャネル上で多重化されたストリームを送信する場合、チャネルは複数のストリームが含まれ、各ストリームに独自の SSRC フィールドがあります。以前のリリースでは、SBC は UDP チャネルの単一 SSRC フィールドのみをサポートできました。したがって、SBC は多重化された SRTP と RTP を送信するエンドポイントのインターワーキングをサポートできませんでした。リリース 3.4S から、SBC は多重化された SRTP または RTP ストリーム内の複数の SSRC フィールドを処理できます。RTP に多重化された RTCP の SRTP サポートとともに、この機能は、RTP ベースおよび SRTP ベースのエンドポイントのインターワーキングを強化します。

グローバル セキュア メディアの設定例

この項では、セキュア メディア パススルー機能のサンプル設定を提供します。

```
Router# configure
Router(config)# sbc mysbc
Router(config-sbc)# sbe
Router(config-sbc-sbe)# secure-media
Router(config-sbc-sbe)# end
```

シグナリングされない詳細レベルのセキュア メディアの設定例

次の設定例は、クライアントとサーバの SIP 隣接を「security trusted-unencrypted」に設定し、CAC テーブル エントリ 1 を発信側と着信側の両方でセキュア メディア用に設定する方法を示します。

（注） この手順では、**caller** コマンドと **callee** コマンドが使用されています。シナリオによっては、**caller** と **callee** のコマンド ペアの代わりに **branch** コマンドを使用できます。**branch** コマンドはリリース 3.5.0 で導入されました。このコマンドの詳細については、「ダイレクト非制限 CAC ポリシーの設定」（P.139）を参照してください。

```
...
cac-policy-set 2
 first-cac-table 1
 cac-table 1
  table-type limit all
  entry 1
   match-value call-update
   caller secure-media
   callee secure-media
   action cac-complete
   exit
 complete
 exit
active-cac-policy-set 2

adjacency sip client
 nat force-off
 security trusted-unencrypted
 signaling-address ipv4 10.10.100.110
 signaling-port 9060
 remote-address ipv4 10.10.100.10 255.255.255.255
 signaling-peer 10.10.100.10
 signaling-peer-port 9060
 attach
adjacency sip server
 nat force-off
 security trusted-unencrypted
 signaling-address ipv4 10.10.100.110
 signaling-port 9070
 remote-address ipv4 10.10.100.10 255.255.255.255
 signaling-peer 10.10.100.10
 signaling-peer-port 9070
 attach
```

次に、CAC ポリシー テーブル中の信頼されない隣接に対して **srtp support allow** コマンドを使用することで、隣接が *untrusted* の場合に、詳細レベルのシグナリングされないセキュア メディアを設定する例を示します。

```
...
```

```
cac-policy-set 2
 first-cac-table 1
 cac-table 1
  table-type limit all
  entry 1
   match-value call-update
   srtp support allow
   caller secure-media
   callee secure-media
   action cac-complete
   exit
 complete
 exit
active-cac-policy-set 2
```

次に、CAC ポリシー セット 2 に関する詳細な情報を表示し、発信側と着信側でセキュア メディアを設定する方法を示します。

```
Router# show sbc asr sbe cac-policy-set 2 detail

SBC Service "asr"

CAC Policy Set 2
  Active policy set: Yes
  Description:
  Averaging period: 60 sec
  First CAC table: 1
  First CAC scope: global
  First CAC prefix length: 4294967256

  Table name: 1
    Description:
    Table type: policy-set                      Total call failures: 0

    Entry 1
    CAC scope:
    CAC scope prefix length: 0
    Action: CAC complete                    Number of calls rejected: 0
    Max calls per scope:    Unlimited       Max call rate per scope: Unlimited
    Max in-call rate:       Unlimited       Max out-call rate:       Unlimited
    Max reg. per scope:     Unlimited       Max reg. rate per scope: Unlimited
    Max channels per scope: Unlimited       Max updates per scope:   Unlimited
    Early media:            Allowed         Early media direction:   Both
    Early media timeout:    None            Transcoder per scope:    Allowed
    Callee Bandwidth-Field: None            Caller Bandwidth-Field:  None
    Media bypass:                     Allowed
    Renegotiate Strategy:             Delta
    Max bandwidth per scope:          Unlimited
    SRTP Transport:                   Trusted-Only (by default)
    Caller hold setting:              Standard
    Callee hold setting:              Standard
    Caller privacy setting:           Never hide
    Callee privacy setting:           Never hide
    Caller voice QoS profile:         Default
    Callee voice QoS profile:         Default
    Caller video QoS profile:         Default
    Callee video QoS profile:         Default
    Caller sig QoS profile:           Default
    Callee sig QoS profile:           Default
    Caller inbound SDP policy:        None
    Callee inbound SDP policy:        None
    Caller outbound SDP policy:       None
    Callee outbound SDP policy:       None
    Caller media disabled:
```

```
    Strip All Answer
  Callee media disabled:
    Strip All Offer
  Caller unsignaled secure media:  Allowed
  Callee unsignaled secure media:  Allowed
  Caller tel-event payload type:   Default
  Callee tel-event payload type:   Default
  Media flag:
    Ignore bandwidth-fields (b=), Telephone Event Interworking
  Restrict codecs to list:         Default
  Restrict caller codecs to list:  Default
  Restrict callee codecs to list:  Default
  Maximum Call Duration:           Unlimited
```

次に、着信側の SIP 隣接「server」の詳細情報の一部を示します。セキュリティ trusted-unencrypted が設定されていることが示されています。

```
Router# show sbc asr sbe adjacencies server detail

SBC Service "asr"
  Adjacency server (SIP)
    Status:                Attached
[snip]
    Security:              Trusted-Unencrypted
[snip]
```

SRTP パススルーの設定例

次に、「srtp transport」変数が隣接の CAC ポリシー セット 1 テーブルに設定されていて、SRTP パススルーを許可する設定を示します。

```
sbc SBE-NODE2-SBE1
   sbe
      cac-policy-set 1
         first-cac-scope global
         first-cac-table STANDARD-LIST-BY-ACCOUNT
         cac-table STANDARD-LIST-BY-ACCOUNT
            table-type limit dst-account
            entry 1
               media-bypass-forbid
                 match-value SIP-CUSTOMER-1
               max-num-calls 100
               max-call-rate 20
               max-bandwidth 1000000 bps
               callee-privacy never
               srtp support allow
               action cac-complete
               exit
            entry 2
               match-value SIP-CUSTOMER-2
               max-num-calls 100
               max-call-rate 20
               max-bandwidth 1000000 bps
               transcode-deny
               max-regs 500
               action cac-complete
               exit
               exit
               complete
      active-call-policy-set 1
```

次に、CAC ポリシー セット 100 のテーブル CAC1 内のエントリを表示し、ポリシーが適用されている隣接で SRTP パススルーを許可するように SRTP トランスポート変数が設定された例を示します。

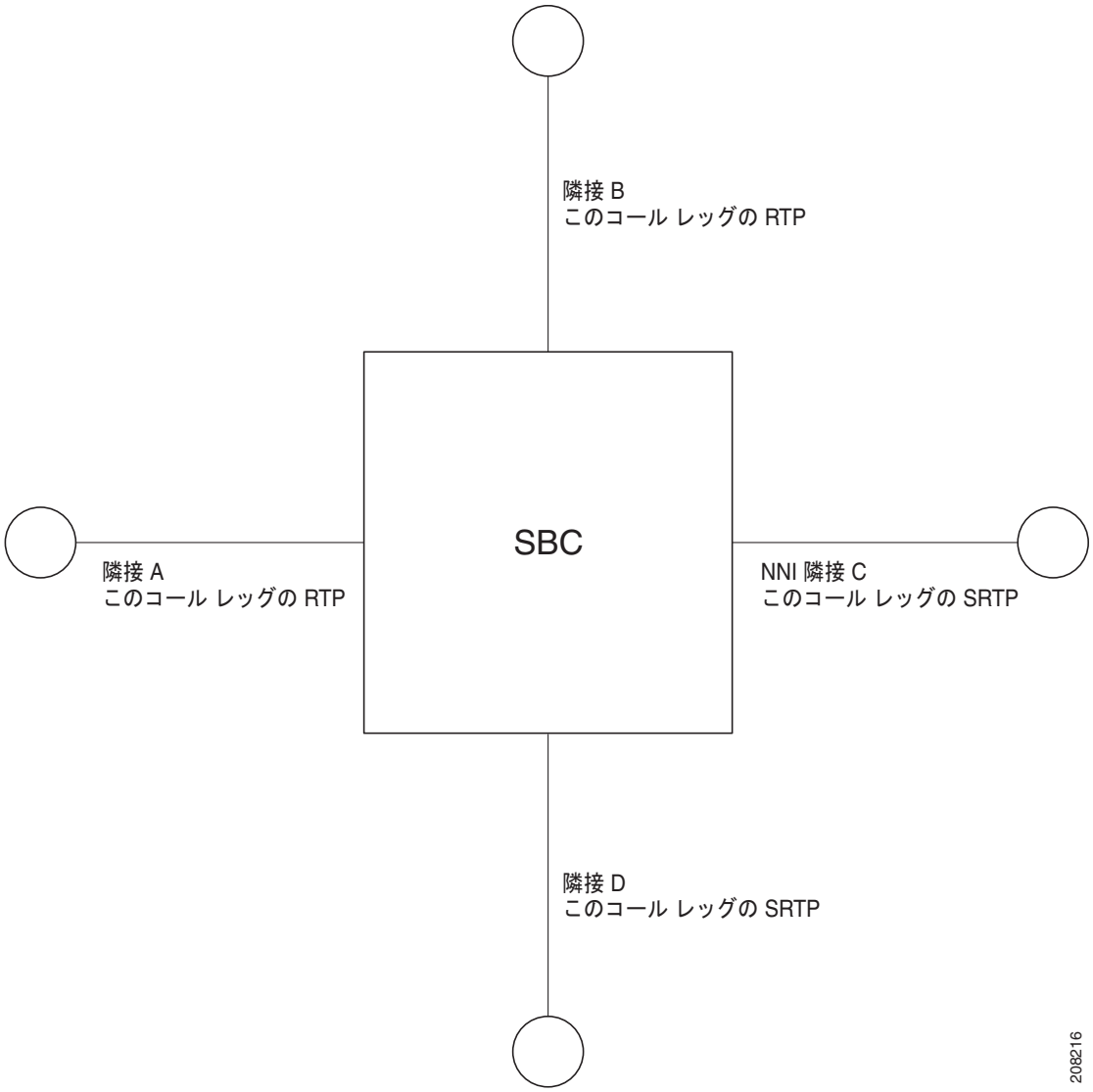
```
Router# show sbc SBC1 sbe cac-policy-set 100 table CAC1 entry 1000

SBC Service "SBC1"
Policy set 100 table CAC1 entry 1000
  Match value               src-adjacency
  Action                    CAC policy complete
  Max calls                 Unlimited
  Max call rate             100
  Max registrations         Unlimited
  Max reg. rate             Unlimited
  Max bandwidth             Unlimited
  Max channels              Unlimited
  Transcoder                Allowed
  Caller privacy setting    Never hide
  Callee privacy setting    Never hide
  Early media               Allowed
  Early media direction     Both
  Early media timeout       0
  Restrict codecs to list   default
  Media bypass              Allowed
  Number of calls rejected by this entry    0
  SRTP Transport            Allowed
```

SRTP と RTP 間のインターワーキングのための CAC ポリシーの設定例

次の例は、RTP と SRTP 間のインターワーキング用にコールの発信側と着信側の CAC ポリシーを設定する方法の詳細を示します。特定の設定を持つ複数のエントリが指定されます。

図 2 は、この例で match-value コマンドで使用する隣接関係を示します。

図 2　　隣接 A、B、C、および D の例




```
configure terminal
sbc SBC1
sbe

cac-policy-set 44
    first-cac-table 44

    cac-table 44
     table-type limit src-adjacency

     entry 1
      match-value A
      srtp support allow
      action next-table 45
      srtp caller forbid
      srtp interworking allow
      srtp media interworking allow
```

```
   entry 2
     match-value B
     srtp support allow
     action next-table 45
     srtp caller forbid
     srtp interworking allow
     srtp media interworking allow

   entry 3
     match-value C
     srtp support allow
     action next-table 45
     srtp caller mandate
     srtp interworking allow
     srtp media interworking allow

entry 4
     match-value D
     srtp support allow
     action next-table 45
     srtp caller mandate
     srtp interworking allow
     srtp media interworking allow

   cac-table 45
    table-type limit dst-adjacency

    entry 1
     match-value A
     srtp support allow
     action cac-complete
     srtp callee forbid
     srtp interworking allow
     srtp media interworking allow

entry 2
     match-value B
     srtp support allow
     action cac-complete
     srtp callee forbid
     srtp interworking allow
     srtp media interworking allow

entry 3
     match-value C
     srtp support allow
     action cac-complete
     srtp callee mandate
     srtp interworking allow
     srtp media interworking allow

entry 4
     match-value D
     srtp support allow
     action cac-complete
     srtp callee mandate
     srtp interworking allow
     srtp media interworking allow

     complete
     end

show sbc sbc1 sbe cac-policy-set 44 detail
```