



トランスコーディングの実装

トランスコーディングは、あるコーデックを使用して符号化されたメディア ストリームを、別のコーデックを使用して符号化されたメディア ストリームに変換する処理です。たとえば、Pulse Code Modulation U-law (PCMU) として符号化されたメディア ストリームを ITU-T G.726-32 として符号化されたメディア ストリームに変換します。

トランスコーディングを設定する主な理由は、外付けのメディア トランスコーディング デバイスを自動的に検出できない場合に、これらのデバイスの機能を設定するためです。トランスコーダの機能であるインバンドの自動検出は、現在サポートされていません。したがって、現行のすべてのリモート トランスコーディング デバイスへの接続はすべて、設定する際にこの手順を実行する必要があります。



(注) 前述の理由が当てはまらない場合は、トランスコーディングの設定をすべて省略できます。

トランスコーディングを設定済みであるかどうかにかかわらず、メディア ゲートウェイは接続が許可されます。設定エラーを回避できるように、Signaling Border Element (SBE) は、Data Border Element (DBE) ではなく、かつトランスコーディングが設定されていないメディア ゲートウェイからインバンド接続を受信した場合、警告をログに記録します。



(注) この機能は、Cisco IOS XE Release 2.5 以降の統合モデルでサポートされます。

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。

SBC トランスコーディング実装の機能履歴

リリース	変更内容
Cisco IOS XE Release 2.5	この機能は、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 3.3S	隣接単位の音声トランスコーディング統計情報機能が、Cisco ASR 1000 シリーズ集約サービス ルータに追加されました。

内容

この章の内容は、次のとおりです。

- 「トランスコーディングの実装の前提条件」(P.858)
- 「トランスコーディングの実装に関する制約事項」(P.858)
- 「メディア ゲートウェイを使用した DTMF インターワーキングの制約事項」(P.859)
- 「トランスコーディングに関する情報」(P.859)
- 「トランスコーディング実装の設定例」(P.869)
- 「検証」(P.871)
- 「隣接単位の音声トランスコーディング統計情報」(P.872)
- 「隣接単位の音声トランスコーディング統計情報機能の設定」(P.872)
- 「メディア ゲートウェイを使用した DTMF インターワーキング」(P.875)

トランスコーディングの実装の前提条件

SBC トランスコーディングを実装するには、次の前提条件を満たす必要があります。

- これらの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。
- 単純なコールの作成に必要な SBE および DBE のすべての設定が、すでに完了済みである必要があります。トランスコーディングの設定は、これらの設定の後に行います。

トランスコーディングの実装に関する制約事項

トランスコーディング機能の実装に関する制約事項は、次のとおりです。

- トランスコーディングが必要な場合、プロシージャをスロー スタートするため、H.323 ファスト スタート コールが廃棄されます。これは、ファスト スタート 要求を拒否した着信側で実現できません。
- H.323-SIP インターワーキング コールに対してトランスコーディングはサポートされません。
- H.323-H.323 インターワーキング コールに対してトランスコーディングはサポートされません。
- H.323 トランスコーディングでサポートされるコーデックは、G.711 (PCMU および PCMA) と G.729 (Annex B 付加および付加なし) だけです。
- 音声トランスコーディングが作動中の場合、SBC を出入りする RFC 2833 インバンド パケット、および他のコール レッグ上のアウトオブバンド SIP INFO メッセージまたは SIP NOTIFY リレー メッセージを含むインターワーキング RFC 2833 パケットの送受信を SBC ではサポートしません。

トランスコーディングを使用する場合、Dual Tone MultiFrequency (DTMF) インターワーキングの制約事項は、次のとおりです。

- コールでトランスコーディングが実行されている場合、シグナリングおよびメディア DTMF インターワーキングはサポートされません。

- 音声トランスコーディングが作動中の場合、SBC を出入りする RFC 2833 インバンド パケット、および他のコール レッグ上のアウトオブバンド SIP INFO メッセージまたは SIP NOTIFY リレー メッセージを含むインターワーキング RFC 2833 パケットの送受信を SBC ではサポートしません。

メディア ゲートウェイを使用した DTMF インターワーキングの制約事項

次に、メディア ゲートウェイを使用した DTMF インターワーキングの制約事項について説明します。

- SBC は、SIP-SIP コールの場合のみ、Cisco MGX 8880 などのトランスコーダの使用をサポートします。トランスコーダを使用した DTMF インターワーキングは H.323 コールではサポートされません。
- SBC は、DTMF をパズスルーできないトランスコーダを使用した DTMF とは相互作用できません。
- Cisco MGX 8880 がトランスコーダとして使用されていない場合は、SIP-SIP コールだけがサポートされます。

トランスコーディングに関する情報

トランスコーディングは、あるコーデックを使用して符号化されたメディア ストリームを、別のコーデックを使用して符号化されたメディア ストリームに変換する処理です。たとえば、PCMU として符号化されたメディア ストリームを G.726-32 として符号化されたメディア ストリームに変換します。

トランスコーディングは、外付けの Digital Signal Processor (DSP; デジタル シグナル プロセッサ) ハードウェアを使用してサポートされます。Cisco MGX 8880 メディア ゲートウェイを使用して、1 つまたは複数の SBC でトランスコーディング機能を提供できます。

SBC がサポートするトランスコーディングは、次の 2 種類です。

- 「拒否後のトランスコーディング」(P.859)
- 「コーデック フィルタリング」(P.862)

拒否後のトランスコーディング

SBC は、コールに対するトランスコーダ サービスの提供が Call Admission Control (CAC; コール アドミッション制御) ポリシー設定によって禁止されない限り、これらのコーデック間でトランスコーディングを必要とするコールでトランスコーディング デバイスを自動的に使用させます。トランスコーディングの必要なコールがセットアップされた場合、SBC は次の段階に沿って進みます。

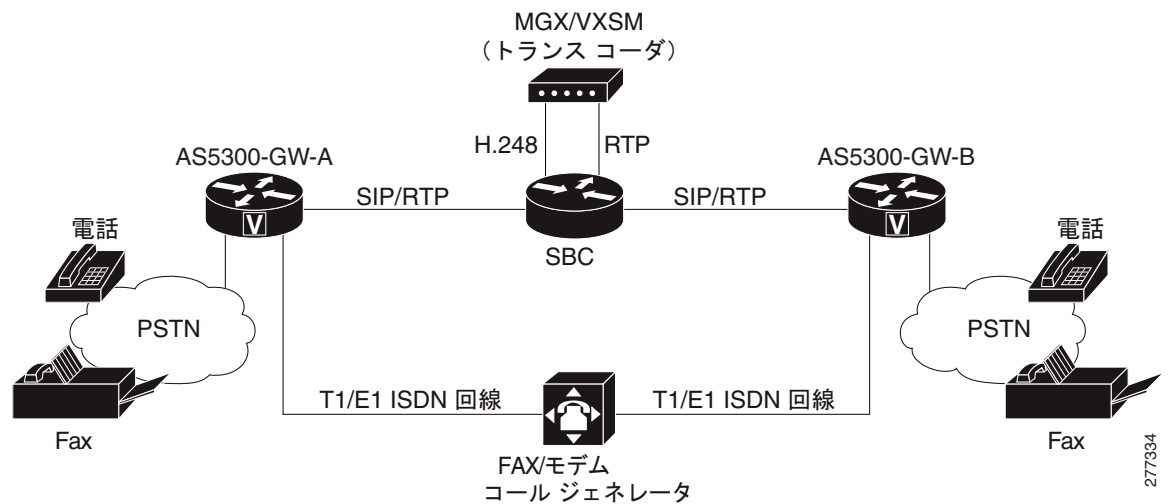
- 発信側エンドポイントからの初回シグナリング要求を受信します。これを受けて、SBC は、初回コール セットアップを発着信のローカル メディア ターミネーション ポイントで実行します。次に、SBC は着信側に向けてセットアップ要求を転送します。
- 着信側エンドポイントから初回要求に受け入れ可能なコーデックがないことを示す応答を受信します。この応答には以下が含まれています。
 - 415 : サポートされないメディア タイプ (SIP)。
 - 488 : ここで受け入れできません (SIP)。
 - H.245 プロトコルの Terminal Capability Exchange 手続きの間に一般的なコーデックの識別に失敗する。

これを受けて、SBC は、着信と発信のローカル メディア ターミネーションの間のメディア パスに挿入したトランスコーダをコールに使用させます。新規要求が着信側エンドポイントに送信され、トランスコーダによって生成された新しいコーデック タイプが示されます。

- ここで、SBE は、着信側エンドポイントで受け入れ可能なコーデックを見つけるまで、トランスコーダでサポートされるコーデックのリストに沿って繰り返し行うことが必要になる場合もあります。これを完了すると、コールが接続されメディアの転送が開始されます。

図 1 には、トランスコーダがネットワーク上に配置されている状態で、トランスコーディングが実行されたコールのメディアが取得したパスを示します。

図 1 トランスコーディング設定



(注)

図 1 では 2 つの DBE を示していますが、トランスコーディングは 1 つの DBE でも実現できます。DBE が 1 つの場合、メディアは DBE を 2 回流れます。1 回めは送信側エンドポイントからトランスコーダへ向かう途中に、2 回めはそのメディアがトランスコーダから受信側エンドポイントに流れるときです。

Session Border Controller (SBC; セッション ボーダー コントローラ) でトランスコーダをプログラムする場合は、トランスコーダを登録する必要があります。トランスコーディング デバイスは、H.248 メディア ゲートウェイとして動作するため、接続先の SBE または SBC の IP アドレスおよびポートを使用して設定する必要があります。SBE または SBC は、H.248 メディア ゲートウェイ コントローラとして動作します。トランスコーダ デバイスの設定方法の注意事項については、トランスコーダ デバイスのマニュアルを参照してください。Cisco MGX 8880 メディア ゲートウェイのマニュアルについては、次の URL から入手できます。

http://www.cisco.com/en/US/docs/switches/wan/mgx/software/mgx_r5.0/data/configuration/guide/scg.html

また、SBE には次の固有の設定が必要です。

- sbe control address ipv4** コマンドおよび **sbe control address h248 port** コマンドを使用して、H.248 コントロール アドレスおよびポートを設定する必要があります。デフォルトで SBE はポート 2944 上に存在します。これは、トランスコーダが接続する必要のあるアドレスおよびポートのことです。

- **sbe media-gateway ipv4** コマンドを使用して、明示的なメディア ゲートウェイを設定する必要があります。明示的なメディア ゲートウェイは、トランスコーダでどのコーデック間の変換が可能かを SBC で把握するために、定義済みのサポート対象コーデックのリストを用意する必要があります。さらにトランスコーダとして識別する必要があります (**sbe media-gateway ipv4 codecs** コマンドおよび **sbe media-gateway ipv4 transcoder** コマンドを使用)。
- **show sbc sbe media-gateway-associations** コマンドを使用して、トランスコーダが SBE に正しく登録されたことを確認できます。正しく登録されていれば、トランスコーダが既知のメディア ゲートウェイのリストに、アクティブなアソシエーションとともに表示されるはずですが。

設定手順については、「拒否後のトランスコーディングの設定」(P.862) を参照してください。

VXSM カードを使用したトランスコーディングされたコールのメディア タイムアウトに関するトラブルシューティングのヒント

Cisco Voice Switch Service Module (VXSM) カードセットを 1 つ以上装備した Cisco MGX 8880 は、メディア ゲートウェイとして動作できます。SBC が、H.248 メディア ゲートウェイとして動作するトランスコーディング デバイスとして Cisco MGX 8880 を使用するネットワークにおいて、メディア タイムアウトが発生したときにトランスコーディングされたコールで正しく動作するようにするには、VISM カードでの追加設定が必要です。

Cisco MGX 8880 メディア ゲートウェイの VISM カードで、次の手順を実行する必要があります。

ステップ 1 次のコマンドを使用して、RTCP 制御をイネーブルにします。

```
InteropMGX.4.VXSM.a > cnfdspparam -control 1
```

ステップ 2 次のコマンドを使用して、RTCP タイマー制御を startRtpOrRtcpPktRcvd に設定します。

```
InteropMGX.4.VXSM.a > cnfdspparam -rtcptm 3
```

ステップ 3 次のコマンドを使用して DSP パラメータのリストを表示し、設定が正しいことを確認します。

```
InteropMGX.4.VXSM.a > dspdspparam
```

```
=====
List DSP Parameters
=====
SID Payload Type           : decimal
RTCP Control                : true                <=== RTCP control enabled
RTCP Interval(milliseconds) : 5000
RTCP Interval Multiplier   : 5
VAD Adaptive                : false
G.711 PLC                   : none
DTMF Power Level (0.1 dBm) : -120
DTMF Power Twist (0.1 dB)  : 0
RTCP Timer Control         : startRtpOrRtcpPktRcvd <=Timer Control properly set
VQM Control                 : disable
RTCPXR Control             : enable
RTCPXR Report Frequency    : 1
VQM Default Minimum Gap    : 16
RTCPXR external R factor   : 127
SES Threshold (ms)         : 50
Voice IPIP mode            : normal
=====
```

VISM カードの詳細については、『Cisco Voice Switch Service Module (VXSM) Configuration Guide Release 5.5』の「VXSM as a Transcoding Gateway」を参照してください。この資料は、次の URL で入手できます。

http://www.cisco.com/en/US/docs/switches/wan/mgx/software/mgx_r5.5/voice/vxsm/configuration/guide/config5.html

コーデック フィルタリング

SBC を使用して、特定のコーデックをホワイトリストに加えることにより、特定のコール、発信側、および着信側で使用できるコーデックを制限できます。初期状態では認識されているすべてのコーデックがホワイトリストに入っています。コール、発信側、着信側のコーデック ホワイトリストに存在しないコーデックが要求された場合、コールはそのまま続行しますが、オファーおよびメディア ゲートの設定から禁止されたコーデックは削除されます。

発信側および着信側のコーデック リストをサポートすることにより、SBC は、よりインテリジェントなトランスコーディングの決定を行うことができます。発信側または着信側のいずれかのエンドポイントでのコーデックのサポートを把握している場合は、発信側、着信側、または両方のリストを CAC ポリシーに設定するのが適切です。ただし、場合によっては、送信元隣接など、他の考慮事項がコーデックの決定に影響することがあります。この場合は、コール単位のコーデック リストを引き続き使用できます。

たとえば、発信側および着信側のコーデック リストを「A と B」に設定した場合、すべてのコールはコーデック A を使用することになります。ただし、コーデック B だけをサポートする中継ネットワーク X（送信元隣接で示されたもの）をコールが通過する場合、ユーザは送信元隣接 X 上で照合する追加のポリシーを持つことができ、コール単位のコーデック リストを B に設定します。ネットワーク X を通過するコールは、このときコーデック B の使用を強制されます。

また、許可された各コーデックに対して最低で許容できる最小パケット化時間の値を設定することにより、各コーデックの最小パケット化時間を制限することもできます。この制限以下のパケット化時間でセッションが要求された場合、コールはそのまま続行しますが、SBC はパケット化時間を設定された最小値まで増やします。

設定手順については、「コーデック フィルタリングのトランスコーディングの設定」(P.865) を参照してください。

拒否後のトランスコーディングの設定

この設定領域では、ユーザは、SBE での管理を必要とすることもあるリモートメディア ゲートウェイのリストの設定を提供します。トランスコーディングが不要な場合、これは必須ではありません。

トランスコーディングを設定する主な理由は、外付けのメディア トランスコーディング デバイスを自動的に検出できない場合に、これらのデバイスの機能を設定するためです。トランスコーダの機能であるインバンドの自動検出は、現在サポートされていません。したがって、現行のすべてのリモート トランスコーディング デバイスへの接続はすべて、設定する際にこの手順を実行する必要があります。



(注)

前述の理由が当てはまらない場合は、トランスコーディングの設定を省略できます。

デフォルトでは、トランスコーディングを設定済みであるかどうかにかかわらず、メディア ゲートウェイは接続が許可されます。設定エラーを回避できるように、SBE は、DBE ではなく、かつトランスコーディングが設定されていないメディア ゲートウェイからインバンド接続を受信した場合、警告をログに記録します。

トランスコーディング実装の基本的な手順は、次のとおりです。

1. SBC で H.248 メディア ゲートウェイ コントローラの IP アドレス、ポート、および転送プロトコルを設定します。メディア ゲートウェイ コントローラがすでに設定されている場合、この手順は必須ではありません。
2. メディア ゲートウェイの IP アドレスを設定します。
3. トランスコードするコーデックを設定します（たとえば、ITU-T G.711 U-law と ITU-T G.729A の間など）。
4. メディア ゲートウェイをトランスコーダとして指定します。
5. SBE をアクティブにします。

このタスクは、SBC のトランスコーディングを実装します。

設定が完了すると、SBC は、**transcode-deny** コマンドを使用して、コールに対するトランスコーダサービスの提供がコール アドミッション制御 (CAC) ポリシー設定によって禁止されない限り、コーデック間でトランスコーディングを必要とするコールでトランスコーディング デバイスを自動的に使用させます（「Cisco Unified Border Element (SP Edition) ポリシーの実装」のコール アドミッション コントロール ポリシー セット、CAC テーブル、およびグローバル CAC ポリシー セットの設定の項を参照）。



(注) H.323 隣接設定では、H.323 FastStart のトランスコードされたコールに対して **h245-tunnel disable** コマンドを使用する必要があります。

手順の概要

1. **configure terminal**
2. **sbc service-name**
3. **sbe**
4. **control address h248 index index-number**
5. **port port-number**
6. **ipv4 ipv4_IP_address**
7. **transport [transport-type]**
8. **exit**
9. **media-gateway ipv4 IPv4-IP-address**
10. **codecs codec-list**
11. **transcoder**
12. **exit**
13. **activate**
14. **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	<code>configure terminal</code> 例： Router# <code>configure terminal</code>	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	<code>sbc service-name</code> 例： Router(config)# <code>sbc mysbc</code>	SBC サービスのモードを開始します。 <code>service-name</code> 引数を使用して、サービスの名前を定義します。
ステップ3	<code>sbe</code> 例： Router(config-sbc)# <code>sbe</code>	SBC サービス内で SBE エンティティのモードを開始します。
ステップ4	<code>control address h248 index index-number</code> 例： Router(config-sbc-sbe)# <code>control address h248 index 0</code>	特定の IPv4 H.248 コントロール アドレスを使用するように SBE を設定します。
ステップ5	<code>port port-number</code> 例： Router(config-sbc-sbe-ctrl-h248)# <code>port 2000</code>	H.248 通信用の特定の IPv4 H.248 ポートを使用するように SBE を設定します。
ステップ6	<code>ipv4 ipv4-IP-address</code> 例： Router(config-sbc-sbe-ctrl-h248)# <code>ipv4 1.1.1.1</code>	特定の IPv4 H.248 コントロール アドレスを使用するように SBE を設定します。
ステップ7	<code>transport [transport-type]</code> 例： Router(config-sbc-sbe-ctrl-h248)# <code>transport udp</code>	H.248 通信の転送タイプを設定します。
ステップ8	<code>exit</code> 例： Router(config-sbc-sbe-ctrl-h248)# <code>exit</code>	現在のコンフィギュレーション モードを終了します。
ステップ9	<code>media-gateway ipv4 IPv4-IP-address</code> 例： Router(config-sbc-sbe)# <code>media-gateway ipv4 10.0.0.1</code>	メディア ゲートウェイを設定します。

	コマンドまたはアクション	目的
ステップ 10	<code>codecs codec-list</code> 例： Router(config-sbc-sbe-mg)# codecs m=audio 1234 RTP/AVP 0 18,a=rtppmap:0 PCMU/8000,a=rtppmap:18 G729A/8000	メディア ゲートウェイでサポートされるコーデックを設定します。メディア ゲートウェイ コーデック コンフィギュレーション モードを開始します。
ステップ 11	<code>transcoder</code> 例： Router(config-sbc-sbe-mg-codecs)# transcoder	メディア ゲートウェイをトランスコーダのサポートを指定して設定します。
ステップ 12	<code>exit</code> 例： Router(config-sbc-sbe-mg-codecs)# exit	メディア ゲートウェイ コーデック コンフィギュレーション モードから <code>sbe</code> コマンド モードのレベルに戻ります。
ステップ 13	<code>activate</code> 例： Router(config-sbc-sbe-mg)# activate	SBE アドレス設定がすべて正常にコミットされると、SBC サービスが開始されます。
ステップ 14	<code>end</code> 例： Router(config-sbc)# end	コンフィギュレーション セッションを終了します。

コーデック フィルタリングのトランスコーディングの設定

次に示すように、コーデック フィルタリングのトランスコーディングを設定します。

手順の概要

1. `configure terminal`
2. `sbc service-name`
3. `sbe`
4. `media-gateway ipv4 IPv4_IP_address`
5. `codecs codec-list`
6. `transcoder`
7. `exit`
8. `cac-policy-set`
9. `first-cac-table`
10. `cac-table`
11. `table-type policy-set`
12. `entry entry-num`
13. `caller-codec-list list-name`

- 14. exit
- 15. exit
- 16. exit
- 17. codec-list *list-name*
- 18. codec *codec-name*
- 19. exit
- 20. 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	<code>configure terminal</code> 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	<code>sbc service-name</code> 例： Router(config)# sbc mysbc	SBC サービスのモードを開始します。 <i>service-name</i> 引数を使用して、サービスの名前を定義しま す。
ステップ3	<code>sbe</code> 例： Router(config-sbc)# sbe	SBC サービス内で SBE エンティティのモードを開始しま す。
ステップ4	<code>media-gateway ipv4 IPv4-IP-address</code> 例： Router(config-sbc-sbe)# media-gateway ipv4 10.0.0.1	メディア ゲートウェイを設定します。
ステップ5	<code>codecs codec-list</code> 例： Router(config-sbc-sbe-mg)# codecs m=audio 1234 RTP/AVP 0 18,a=rtpmap:0 PCMU/8000,a=rtpmap:18 G729A/8000	メディア ゲートウェイでサポートされるコーデックを設定 します。
ステップ6	<code>transcoder</code> 例： Router(config-sbc-sbe-mg-codecs)# transcoder	メディア ゲートウェイをトランスコーダのサポートを指定 して設定します。
ステップ7	<code>exit</code> 例： Router(config-sbc-sbe-mg-codecs)# exit	メディア ゲートウェイ コンフィギュレーション モードを 終了します。

	コマンドまたはアクション	目的
ステップ 8	cac-policy-set 例： Router(config-sbc-sbe)# cac-policy-set 1	CAC ポリシー サブモードを開始します。
ステップ 9	first-cac-table 例： Router(config-sbc-sbe-cacpolicy)# first-cac-table 1	最初のアドミSSION コントロール テーブルを作成または設定します。
ステップ 10	cac-table 例： Router(config-sbc-sbe-cacpolicy)# cac-table 1	アドミSSION コントロール テーブルを作成または設定します。
ステップ 11	table-type policy-set 例： Router(config-sbc-sbe-cacpolicy-cactable)# table-type policy-set	コール アドミSSION 制御 (CAC) テーブルのポリシー セット テーブル タイプを設定します。
ステップ 12	entry 例： Router(config-sbc-sbe-cacpolicy-cactable)# entry 1	テーブル内のエントリを作成または変更します。
ステップ 13	caller-codec-list list-name 例： Router(config-sbc-sbe-cacpolicy-cactable-entry) # caller-codec-list my_codecs	コールの発信側レグが使用を許可されているコーデックを一覧表示します。
ステップ 14	exit 例： Router(config-sbc-sbe-cacpolicy-cactable-entry) # exit	CAC テーブル エントリ コンフィギュレーション モードを終了します。
ステップ 15	exit 例： Router(config-sbc-sbe-cacpolicy-cactable)# exit	CAC テーブル コンフィギュレーション モードを終了します。
ステップ 16	exit 例： Router(config-sbc-sbe-cacpolicy)# exit	CAC ポリシー コンフィギュレーション モードを終了します。
ステップ 17	codec-list list-name 例： Router(config-sbc-sbe)# codec-list my_codecs	コーデック リストを作成し、コーデック リスト コンフィギュレーション 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 18	codec <i>codec-name</i> 例 : Router(config-sbc-sbe-codec-list)# codec PCMU	コーデックをコーデック リストに追加します。
ステップ 19	exit 例 : Router(config-sbc-sbe-codec-list)# exit	コーデック リスト コンフィギュレーション モードを終了します。
ステップ 20	end 例 : Router(config-sbc-sbe)# end	コンフィギュレーション セッションを終了します。

トランスコーディング実装の設定例

以下は、拒否後のトランスコーディングの設定例です。

```
Router(config)# sbc mySbc
Router(config-sbc)# sbe
Router(config-sbc-sbe)# control address h248 index 1
Router(config-sbc-sbe-ctrl-h248)# port 2000
Router(config-sbc-sbe-ctrl-h248)# ipv4 88.88.133.2
Router(config-sbc-sbe-ctrl-h248)# transport udp
Router(config-sbc-sbe-ctrl-h248)# exit
Router(config-sbc-sbe)# media-gateway ipv4 10.0.0.1
Router(config-sbc-sbe-mg)# codecs m=audio 1234 RTP/AVP 0 18,a=rtpmap:0
PCMU/8000,a=rtpmap:18 G729A/8000
Router(config-sbc-sbe-mg-codecs)# transcoder
Router(config-sbc-sbe-mg-codecs)# exit
Router(config-sbc-sbe-mg)# activate
Router(config-sbc)# end
```

以下は、コーデック フィルタリングのトランスコーディングの例です。

```
Router(config)# ip route 10.0.20.33 255.255.255.255 10.130.10.33
Router(config)# ip route 0.0.0.0 0.0.0.0 10.74.50.114
Router(config)# ip route 0.0.0.0 0.0.0.0 10.130.10.1

Router(config)# snmp-server community cisco group Network-Monitor
Router(config)# snmp-server community public group Network-Monitor
Router(config)# snmp-server community private group Network-Monitor

Router# configure terminal
Router(config)# sbc sbc-11
Router(config-sbc)# sbe
Router(config-sbc-sbe)# media-gateway ipv4 10.100.181.2
Router(config-sbc-sbe-mg)# codecs m=audio 20000 RTP/AVP 0 8 18,a=rtpmap:0
PCMU/8000,a=rtpmap:8 PCMA/8000,a=rtpmap:18 G729/8000
Router(config-sbc-sbe-mg)# transcoder

Router(config-sbc-sbe)# control address h248 index 1
Router(config-sbc-sbe-ctrl-h248)# ipv4 10.130.10.4
Router(config-sbc-sbe-ctrl-h248)# transport udp

Router(config-sbc-sbe)# adjacency sip SIPP81
Router(config-sbc-sbe-adj-sip)# nat force-off
Router(config-sbc-sbe-adj-sip)# preferred-transport udp
Router(config-sbc-sbe-adj-sip)# redirect-mode pass-through
Router(config-sbc-sbe-adj-sip)# authentication nonce timeout 300
Router(config-sbc-sbe-adj-sip)# signaling-address ipv4 10.130.10.4
Router(config-sbc-sbe-adj-sip)# signaling-port 5060
Router(config-sbc-sbe-adj-sip)# remote-address ipv4 10.0.244.81 255.255.255.255
Router(config-sbc-sbe-adj-sip)# signaling-peer 10.0.244.81
Router(config-sbc-sbe-adj-sip)# signaling-peer-port 5060
Router(config-sbc-sbe-adj-sip)# db-location-id 0
Router(config-sbc-sbe-adj-sip)# reg-min-expiry 3000
Router(config-sbc-sbe-adj-sip)# attach

Router(config-sbc-sbe)# adjacency sip SIPP91
Router(config-sbc-sbe-adj-sip)# nat force-off
Router(config-sbc-sbe-adj-sip)# preferred-transport udp
Router(config-sbc-sbe-adj-sip)# redirect-mode pass-through
Router(config-sbc-sbe-adj-sip)# authentication nonce timeout 300
Router(config-sbc-sbe-adj-sip)# signaling-address ipv4 10.130.10.4
```

```

Router (config-sbc-sbe-adj-sip) # signaling-port 5060
Router (config-sbc-sbe-adj-sip) # remote-address ipv4 10.0.244.91 255.255.255.255
Router (config-sbc-sbe-adj-sip) # signaling-peer 10.0.244.91
Router (config-sbc-sbe-adj-sip) # signaling-peer-port 5060
Router (config-sbc-sbe-adj-sip) # db-location-id 0
Router (config-sbc-sbe-adj-sip) # reg-min-expiry 3000
Router (config-sbc-sbe-adj-sip) # attach

Router (config-sbc-sbe) # sip inherit profile preset-core

Router (config-sbc-sbe) # cac-policy-set 1
Router (config-sbc-sbe-cacpolicy) # first-cac-table table
Router (config-sbc-sbe-cacpolicy) # first-cac-scope call
Router (config-sbc-sbe-cacpolicy) # averaging-period 60
Router (config-sbc-sbe) # cac-table table
Router (config-sbc-sbe-cacpolicy-cactable) # match-type adjacency
Router (config-sbc-sbe-cacpolicy-cactable) # entry 1
Router (config-sbc-sbe-cacpolicy-cactable-entry) # match-value SIPP81
Router (config-sbc-sbe-cacpolicy-cactable-entry) # action cac-complete
Router (config-sbc-sbe-cacpolicy-cactable-entry) # max-bandwidth 64009 Gbps
Router (config-sbc-sbe-cacpolicy-cactable-entry) # max-updates 4294967295
Router (config-sbc-sbe-cacpolicy-cactable-entry) # max-channels 4294967295
Router (config-sbc-sbe-cacpolicy-cactable-entry) # early-media-type full-duplex
Router (config-sbc-sbe-cacpolicy-cactable-entry) # early-media-timeout 0
Router (config-sbc-sbe-cacpolicy-cactable-entry) # caller-codec-list allow711u
Router (config-sbc-sbe-cacpolicy-cactable-entry) # callee-privacy never
Router (config-sbc-sbe-cacpolicy-cactable-entry) # caller-privacy never
Router (config-sbc-sbe-cacpolicy-cactable-entry) # callee-hold-setting standard
Router (config-sbc-sbe-cacpolicy-cactable-entry) # caller-hold-setting standard
Router (config-sbc-sbe-cacpolicy-cactable) # entry 2
Router (config-sbc-sbe-cacpolicy-cactable-entry) # match-value SIPP91
Router (config-sbc-sbe-cacpolicy-cactable-entry) # action cac-complete
Router (config-sbc-sbe-cacpolicy-cactable-entry) # max-bandwidth 64009 Gbps
Router (config-sbc-sbe-cacpolicy-cactable-entry) # max-updates 4294967295
Router (config-sbc-sbe-cacpolicy-cactable-entry) # max-channels 4294967295
Router (config-sbc-sbe-cacpolicy-cactable-entry) # early-media-type full-duplex
Router (config-sbc-sbe-cacpolicy-cactable-entry) # early-media-timeout 0
Router (config-sbc-sbe-cacpolicy-cactable-entry) # callee-codec-list allowg729
Router (config-sbc-sbe-cacpolicy-cactable-entry) # callee-privacy never
Router (config-sbc-sbe-cacpolicy-cactable-entry) # caller-privacy never
Router (config-sbc-sbe-cacpolicy-cactable-entry) # callee-hold-setting standard
Router (config-sbc-sbe-cacpolicy-cactable-entry) # caller-hold-setting standard
Router (config-sbc-sbe-cacpolicy-cactable-entry) # complete

Router (config-sbc-sbe) # active-cac-policy-set 1

Router (config-sbc-sbe) # retry-limit 3

Router (config-sbc-sbe) # call-policy-set 1
Router (config-sbc-sbe-rtgpolicy) # first-call-routing-table table
Router (config-sbc-sbe-rtgpolicy-rtgtable) # entry 1
Router (config-sbc-sbe-rtgpolicy-rtgtable-entry) # action complete
Router (config-sbc-sbe-rtgpolicy-rtgtable-entry) # dst-adjacency SIPP91
Router (config-sbc-sbe-rtgpolicy-rtgtable-entry) # match-address 318X
Router (config-sbc-sbe-rtgpolicy-rtgtable-entry) # prefix
Router (config-sbc-sbe-rtgpolicy-rtgtable-entry) # complete

Router (config-sbc-sbe) # active-call-policy-set 1

Router (config-sbc-sbe) # sip max-connections 2
Router (config-sbc-sbe) # sip timer
Router (config-sbc-sbe-sip-tmr) # tcp-idle-timeout 120000
Router (config-sbc-sbe-sip-tmr) # tls-idle-timeout 3600000
Router (config-sbc-sbe-sip-tmr) # udp-response-linger-period 32000

```

```

Router(config-sbc-sbe-sip-tmr) # udp-first-retransmit-interval 500
Router(config-sbc-sbe-sip-tmr) # udp-max-retransmit-interval 4000
Router(config-sbc-sbe-sip-tmr) # invite-timeout 180

Router (config-sbc-sbe) # codec-list allow711u
Router (config-sbc-sbe-codec-list) # codec PCMU

Router (config-sbc-sbe) # codec-list allowg729
Router (config-sbc-sbe-codec-list) # codec G729

Router (config-sbc-sbe) # h323
Router (config-sbc-sbe-h323) # ras timeout arq 5000
Router (config-sbc-sbe-h323) # ras retry arq 2
Router (config-sbc-sbe-h323) # ras timeout brq 3000
Router (config-sbc-sbe-h323) # ras retry brq 2
Router (config-sbc-sbe-h323) # ras timeout drq 3000
Router (config-sbc-sbe-h323) # ras retry drq 2
Router (config-sbc-sbe-h323) # ras timeout grq 5000
Router (config-sbc-sbe-h323) # ras retry grq 2
Router (config-sbc-sbe-h323) # ras timeout rrq 3000
Router (config-sbc-sbe-h323) # ras retry rrq 2
Router (config-sbc-sbe-h323) # ras rrq ttl 60
Router (config-sbc-sbe-h323) # ras timeout urq 3000
Router (config-sbc-sbe-h323) # ras retry urq 1
Router (config-sbc-sbe-h323) # h225 timeout proceeding 10000
Router (config-sbc-sbe-h323) # h225 timeout establishment 180000
Router (config-sbc-sbe-h323) # h225 timeout setup 4000
Router (config-sbc-sbe-h323) # ras rrq keepalive 45000

Router (config-sbc-sbe) # h323
Router (config-sbc-sbe-adj-h323) # adjacency timeout 30000

Router (config-sbc-sbe) # blacklist
Router (config-sbc-sbe-blacklist) # global

Router (config-sbc-sbe) # blacklist
Router (config-sbc-sbe-blacklist) # address-default

Router (config-sbc-sbe) # redirect-limit 2
Router (config-sbc-sbe) # deact-mode normal
Router (config-sbc-sbe) # activate

Router (config-sbc) # dbe
Router (config-sbc-dbe) # location-id 0
Router (config-sbc-dbe) # media-timeout 360
Router (config-sbc-dbe) # deact-mode normal
Router (config-sbc-dbe) # activate

```

検証

次の **show sbc sbe media-gateway-associations** コマンドを使用して、アクティブなアソシエーションとともに既知のメディア ゲートウェイのリストを表示し、動作を検証します。

次に、SBC およびメディア通信の例を示します。

```

Router# show sbc slt-n2 sbe media-gateway-associations

SBC Service "slt-n2"
  Media gateway 192.169.125.1:2944
  Gateway Protocol = megaco
  Transport Protocol = UDP

```

	Local Address	= 22.46.0.11:2944			
	Sent	Received	Failed	Retried	
Requests	117	2	1	7	
Replies	2	116	-	0	

隣接単位の音声トランスコーディング統計情報

隣接単位の音声トランスコーディング統計情報機能は、グローバル レベルおよび隣接単位で、音声コールに関するトランスコーディング統計情報を提供します。この機能は、トランスコーディング機能を提供するカード（DSP カードなど）の消費量を分析します。

トランスコーディングの統計情報には、次の情報が含まれています。

- いくつかのサマリー期間における、各コーデック ペアのアクティブなトランスコーディングメディア ストリームの数。グローバル レベルおよび隣接単位で提供されます。統計情報には、対応するコーデック ペアの High Water Mark (HWM; 最高水準点) も含まれます。
- グローバル レベルおよび隣接単位の両方について、アクティブなトランスコーディング コールの数の一覧が表示されます。統計情報は、コーデック ペアのリストで、グローバル レベルおよび隣接単位の両方について示されます。
- 統計情報には、次のコーデック名が提示されます。トランスコーディング コールで他のコーデックを使用している場合は、コーデック名は「*Other*」として表示されます。
 - G711A
 - G711U
 - G729
 - GSM
 - T38
 - CLEAR


隣接単位の音声トランスコーディング統計情報機能の設定

ここでは、隣接単位の音声トランスコーディング統計情報機能を設定し、グローバル レベルおよび隣接単位でトランスコーディング統計情報をリストする方法を説明します。

手順の概要

1. **configure terminal**
2. **sbc *sbc-name***
3. **sbe**
4. **transcoding-stats enable**
5. **end**
6. **show sbc *sbc-name* sbe transcoding-stats {global | adjacency *adjacency-name*} {current15mins | current5mins | currentday | currenthour | current-indefinite | previous15mins | previous5mins | previousday | previoushour}**
7. **clear sbc *sbc-name* sbe transcoding-stats [global | adjacency *adjacency-name*] [all | current-indefinite]**

手順の詳細

	コマンドまたはアクション	目的
ステップ1	<code>configure terminal</code> 例： Router# <code>configure terminal</code>	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	<code>sbc sbc-name</code> 例： Router(config)# <code>sbc mySBC</code>	SBC サービス モードを開始します。 <code>sbc-name</code> 引数を使用して、サービスの名前を定義します。
ステップ3	<code>sbe</code> 例： Router(config-sbc)# <code>sbe</code>	SBC サービス内で SBE エンティティ モードを開始します。
ステップ4	<code>transcoding-stats enable</code> 例： Router(config-sbc-sbe)# <code>no transcoding-stats enable</code>	SBC のトランスコーディング関連統計情報をイネーブルまたはディセーブルにします。 次の警告が発行され、ユーザは「y (Yes)」または「n (No)」を選択してトランスコーディング統計情報をイネーブルまたはディセーブルにする必要があります。  警告 このコマンドによって SBC が再アクティブ化され、既存のコールが影響を受けます。 デフォルトで、SBC のトランスコーディング関連統計情報はイネーブルになっています。
ステップ5	<code>end</code> 例： Router(config-sbc-sbe)# <code>end</code>	SBE エンティティ モードを終了し、特権 EXEC モードを開始します。

コマンドまたはアクション	目的
<p>ステップ6</p> <pre>show sbc sbc-name sbe transcoding-stats {global adjacency adjacency-name} {current15mins current5mins currentday currenthour current-indefinite previous15mins previous5mins previousday previoushour}</pre> <p>例:</p> <pre>Router# show sbc mySBC sbe transcoding-stats adjacency SIP current5mins</pre>	<p>必要なスコープおよびサマリー期間の音声トランスコーディング統計情報をリストします。</p> <ul style="list-style-type: none"> • adjacency : 指定した隣接のトランスコーディング統計情報をリストします。 • global : SBC 全体を対象としたグローバル レベルの統計情報をリストします。 • current15mins : 現在の 15 分間の統計情報をリストします。 • current5mins : 現在の 5 分間の統計情報をリストします。 • currentday : 当日の深夜からの統計情報をリストします。 • currenthour : 現在の時間 (1 時間) の統計情報をリストします。 • currentindefinite : 明示的なりセットが最後に実行された後の統計情報をリストします。 • previous15mins : 過去 15 分間の統計情報をリストします。 • previous5mins : 過去 5 分間の統計情報をリストします。 • previousday : 前日の統計情報をリストします。 • previoushour : 過去 1 時間の統計情報をリストします。
<p>ステップ7</p> <pre>clear sbc sbc-name sbe transcoding-stats [global adjacency adjacency-name] [all current-indefinite]</pre> <p>例:</p> <pre>Router# clear sbc mySBC sbe adjacency SIP all</pre>	<p>すべてまたは現在の、未確定のサマリー期間のトランスコーディングの統計情報をクリアします。</p> <ul style="list-style-type: none"> • adjacency : 隣接の統計情報をクリアします。 • global : グローバルなトランスコーディングの統計情報をクリアします。 • all : すべてのサマリー期間の統計情報をクリアします。 • currentindefinite : 現在未確定の期間についてのみ、統計情報をクリアします。

次に、**show sbc sbe transcoding-stats adjacency current15mins** コマンドの出力例を示します。

```
Router# show sbc mySBC sbe transcoding-stats adjacency SIP current15mins
```

Codec1	Codec2	Transcoded	Stream	HWM of TranscodedStream	Last Reset
G711A	G711U	4		10	2010/09/10 19:27:15

メディア ゲートウェイを使用した DTMF インターワーキング

SBC は MGX 8880 などのメディア ゲートウェイ スイッチを使用したインバンド DTMF インターワーキングをイネーブルにします。Cisco MGX 8880 は、次のシナリオにおいて、DTMF インターワーキングで使用されます。

- トランスコーダとして使用。メディア プレーンとシグナリング間の DTMF がサポートされます。
- インバンド DTMF のエクストラクタまたはインジェクタとして使用。

SBC は、次の 2 種類のメディア プレーン DTMF をサポートします。

- RFC2833 (電話イベント)。
- インバンド DTMF : G.711 などの DTMF インバンド オーディオ ストリーム。インバンド DTMF をサポートするため、MGX は次のタスクを実行します。
 - オーディオ ストリームを監視します。
 - DTMF シグナルを取得します。
 - DTMF シグナルを音声帯域にレポートまたは注入します。または、その逆の処理を行います。

トランスコーダとして MGX を使用する DTMF インターワーキング

SBC が MGX などの外部トランスコーダを使用する場合、DTMF インターワーキングは次の場合にサポートされます。

- コールでのメディアとシグナリング間。
- いずれの場合も、トランスコーディングとコーデック リストを使用してプロビジョニングされるトランスコーディングのネゴシエーションが行われます。
- サポートされるメディア形式 (RFC2833 など) とサポートされる SIP シグナリング形式 (INFO など) の間。

インバンド DTMF サポート (トランスコーダを使用しないインターワーキング)

SBC は、インバンド DTMF トーンが監視されているときを示すコールまたは隣接ポリシーをサポートします。インバンド DTMF トーンの監視は、強制的に行うこともできますし、他の DTMF サポートがない場合にオプションとして実行することもできます。

SBC は、サポートされる 3 つの DTMF 形式のうち、2 つの形式間のインターワーキングをイネーブルにします。

- インバンド
- RFC 2833 電話イベント
- シグナリング。

フェールオーバーが発生した場合、DTMF インターワーキングオプションを使用するアクティブなコールは保護され、インターワーキング機能は復元できるように保持されます。

SBC は、インバンド DTMF と互換性のあるコーデック ネゴシエーションを強制的に実行するように隣接単位のオプションを提供します (他の方法で DTMF の送受信を行うことができない場合)。

インバンド DTMF インターワーキングの設定

インバンド DTMF インターワーキングを設定するには、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 service-name**
3. **sbc**
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 {policy-set | limit {list of limit tables}}**
9. **entry entry-id**
10. **cac-scope {list of scope options}**
11. **callee inband-dtmf-mode {always | inherit | maybe | never}**
12. **caller inband-dtmf-mode {always | inherit | maybe | never}**
13. **complete**
14. **active-call-policy-set policy-set-id**
15. **end**
16. **show sbc service-name sbc 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 service-name 例: Router(config)# sbc mysbc	SBC サービスのサブ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3	<p><code>sbe</code></p> <p>例： Router(config-sbc)# sbe</p>	SBC サービス内で SBE エンティティのサブモードを設定します。
ステップ4	<p><code>cac-policy-set policy-set-id</code></p> <p>例： Router(config-sbc-sbe)# cac-policy-set 1</p>	CAC ポリシーのサブモードを開始します。
ステップ5	<p><code>first-cac-scope scope-name</code></p> <p>例： Router(config-sbc-sbe-cacpolicy)# first-cac-scope global</p>	<p>ポリシーのアドミSSION コントロール ステージを実行する際に制限の定義を開始する範囲を設定します。</p> <p>(注) <code>first-cac-scope</code> 定義が該当するのは、first-cac-table コマンドで設定されたテーブルタイプが制限テーブルの場合だけです。この場合、first-cac-table のスコープは、first-cac-scope により決定されます。first-cac-table がポリシー セットテーブルの場合、first-cac-scope は無視され、デフォルトはグローバルになります。</p> <p><code>scope-name</code> 引数には制限が最初に定義されるべき範囲を設定します。有効な <code>scope-name</code> 値は次のとおりです。</p> <ul style="list-style-type: none"> • adj-group • call • category • dst-account • dst-adj-group • dst-adjacency • dst-number • global • src-account • src-adj-group • src-adjacency • src-number <p>機能は、個々の隣接ごとに行う場合と同じように、CAC 設定から隣接グループごとにイネーブルまたはディセーブルにできます。</p>
ステップ6	<p><code>first-cac-table table-name</code></p> <p>例： Router(config-sbc-sbe-cacpolicy)# first-cac-table first_policy_table</p>	<p>ポリシーのアドミSSION コントロール ステージを実行するときに処理する最初のポリシー テーブルの名前を設定します。</p>

コマンドまたはアクション	目的
<p>ステップ7 <code>cac-table table-name</code></p> <p>例 : Router(config-sbc-sbe-cacpolicy)# cac-table first_policy_table</p>	<p>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 テーブルを設定するためのモードを開始します。必要な場合はテーブルを作成します。</p>
<p>ステップ8 <code>table-type {policy-set limit {list of limit tables}}</code></p> <p>例 : Router(config-sbc-sbe-cacpolicy-cactable)# table-type policy-set</p>	<p>SBE ポリシー セットのコンテキスト内で、CAC テーブルのテーブル タイプを設定します。</p> <p><i>list of limit tables</i> 引数は、テーブル内にあるエントリの <i>match-value</i> フィールドの構文を制御します。有効な <i>list of limit tables</i> の値は次のとおりです。</p> <ul style="list-style-type: none"> • account : アカウント名と比較します。 • adj-group : 隣接グループ名と比較します。 • adjacency : 隣接名と比較します。 • 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と比較します。 • dst-account : 宛先アカウント名と比較します。 • dst-adj-group : 宛先隣接グループ名と比較します。 • dst-adjacency : 宛先隣接の名前と比較します。 • dst-prefix : 着信ディジット スtringの先頭と比較します。 • 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と比較します。 • src-adj-group : 送信元隣接グループ名と比較します。 • src-adjacency : 送信元隣接名と比較します。 • src-prefix : 発番号Stringの先頭と比較します。 <p>(注) 制限テーブルでは、イベント、メッセージ、コールが1つのエントリだけに一致します。</p> <p>機能は、個々の隣接ごとに行う場合と同じように、CAC 設定から隣接グループごとにイネーブルまたはディセーブルにできます。adj-group テーブル タイプは、送信元または宛先の隣接グループで一致します。</p> <p>policy-set キーワードが指定されている場合、cac-scope コマンドを使用して、CAC ポリシー セット テーブル内で制限が適用される、各エントリ内の範囲を設定します。</p> <p>(注) ポリシー セット テーブルでは、イベント、コール、メッセージは、このテーブルのすべてのエントリに適用されます。</p>

コマンドまたはアクション	目的
<p>ステップ9 <code>entry entry-id</code></p> <p>例： Router(config-sbc-sbe-cacpolicy-cactable)# <code>entry 1</code></p>	<p>アドミッション コントロール テーブル内のエントリを作成または変更するモードを開始します。</p>
<p>ステップ10 <code>cac-scope {list of scope options}</code></p> <p>例： Router(config-sbc-sbe-cacpolicy-cactable-entry) # <code>cac-scope call</code></p>	<p>ポリシー セット テーブルで制限を適用する各エントリ内の範囲を設定します。</p> <p>codec-restrict-to-list コマンドを使用する場合は、コール単位だけにスコープを設定できます。</p> <p><i>list of scope options</i> : イベント照合に使用される次のいずれかのストリングを指定します。</p> <ul style="list-style-type: none"> • account : 同じアカウントからのイベント。 • adjacency : 同じ隣接からのイベント。 • adj-group : 同じ隣接グループのメンバーからのイベント。 • call : スコープの限界は単一コール単位です。 • category : 同じカテゴリを持つイベント。 • dst-account : 同じアカウントに送信されるイベント。 • dst-adj-group : 同じ隣接グループに送信されるイベント。 • dst-adjacency : 同じ隣接到送信されるイベント。 • dst-number : 同じ宛先を持つイベント。 • global : スコープの限界はグローバルです。 • src-account : 同じアカウントからのイベント。 • src-adj-group : 同じ隣接グループからのイベント。 • src-adjacency : 同じ隣接からのイベント。 • src-number : 同じ送信元番号を持つイベント。 • sub-category : このスコープで指定される制限は、同じ加入者カテゴリのメンバーとの間で送受信されるすべてのイベントに適用されます。 • sub-category-pfx : このスコープで指定される制限は、同じ加入者カテゴリ プレフィックスのメンバーとの間で送受信されるすべてのイベントに適用されます。 • subscriber : このスコープで指定される制限は、個々の加入者間で送受信されるすべてのイベントに適用されます。 <p>レジストラ サーバに登録されているデバイス。</p>

	コマンドまたはアクション	目的
ステップ 11	<pre>callee inband-dtmf-mode {always inherit maybe never}</pre> <p>例 : Router(config-sbc-sbe-cacpolicy-cactable-entry) # callee inband-dtmf-mode always</p>	<p>着信側の DTMF インバンド モードを設定するには、CAC テーブル コンフィギュレーション モードで callee inband-dtmf-mode コマンドを使用します。着信側の DTMF インバンド モードの設定を解除するには、このコマンドの no 形式を使用します。</p> <p>callee inband-dtmf-mode には、次のいずれかのストリングを指定します。</p> <ul style="list-style-type: none"> • always : インバンド DTMF トーンはエンドポイントで常に使用されています。 • inherit : エンドポイントのインバンド DTMF モードは CAC エントリの影響を受けません。 • maybe : DTMF の代替形式が使用されていることがシグナリングによって示されない限り、インバンド DTMF トーンはエンドポイントで使用されます。 • never : エンドポイントは、インバンド DTMF を使用しません。
ステップ 12	<pre>caller inband-dtmf-mode {always inherit maybe never}</pre> <p>例 : Router(config-sbc-sbe-cacpolicy-cactable-entry) # caller inband-dtmf-mode never</p>	<p>発信側の DTMF インバンド モードを設定するには、CAC テーブル コンフィギュレーション モードで caller inband-dtmf-mode コマンドを使用します。発信側の DTMF インバンド モードの設定を解除するには、このコマンドの no 形式を使用します。</p> <p>caller inband-dtmf-mode には、次のいずれかのストリングを指定します。</p> <ul style="list-style-type: none"> • always : インバンド DTMF トーンはエンドポイントで常に使用されています。 • inherit : エンドポイントのインバンド DTMF モードは CAC エントリの影響を受けません。 • maybe : DTMF の代替形式が使用されていることがシグナリングによって示されない限り、インバンド DTMF トーンはエンドポイントで使用されます。 • never : エンドポイントは、インバンド DTMF を使用しません。
ステップ 13	<pre>complete</pre> <p>例 : Router(config-sbc-sbe-cacpolicy)# complete</p>	CAC ポリシーを完了します。
ステップ 14	<pre>active-call-policy-set policy-set-id</pre> <p>例 : Router(config-sbc-sbe-cacpolicy)# cac-policy-set global 1</p>	SBE エンティティ内でアクティブなルーティング ポリシー セットを設定します。

	コマンドまたはアクション	目的
ステップ 15	<code>end</code> 例： Router(config-sbc-sbe-cacpolicy)# end	cac-policy-set コンフィギュレーション モードを終了して、特権 EXEC モードを開始します。
ステップ 16	<code>show sbc service-name sbe cac-policy-set id table name entry entry</code> 例： Router# show sbc mysbc sbe cac-policy-set 1 table standard_policy_list entry 1	CAC ポリシー テーブルの特定エントリに関する詳細情報を表示します (DTMF インターワーキングなど)。

インバンド DTMF をサポートするコーデックの設定

インバンド DTMF をサポートするようにコーデックを設定するには、次の作業を実行します。

手順の概要

1. `configure terminal`
2. `sbc service-name`
3. `sbe`
4. `codec system system-name id`
5. `options {none | transrate | transcode | inband-dtmf}`
6. `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	<code>configure terminal</code> 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	<code>sbc service-name</code> 例： Router(config)# sbc mysbc	SBC サービスのサブモードを開始します。
ステップ 3	<code>sbe</code> 例： Router(config-sbc)# sbe	SBC サービス内で SBE エンティティのサブモードを設定します。
ステップ 4	<code>codec system system id</code> 例： Router(config-sbc-sbe)# codec system GSM id 3	アナログからデジタルへのシステム コーデック (enCOder/DECOder) の名前を指定し、コーデック定義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	<p><code>options {none transrate transcode inband-dtmf}</code></p> <p>例： <pre>Router(config-sbc-sbe-codec-def)# options inband-dtmf</pre></p>	<p>音声インバンド DTMF をサポートするコーデックを設定します。オプションの値は、次のとおりです。</p> <ul style="list-style-type: none"> • none • transrate • transcode • inband-dtmf
ステップ6	<p><code>end</code></p> <p>例： <pre>Router(config-sbc-sbe-cacpolicy)# end</pre></p>	<p><code>cac-policy-set</code> コンフィギュレーション モードを終了して、特権 EXEC モードを開始します。</p>