

Cisco IOS および IOS XE コールルーティングについての説明

内容

[はじめに](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[背景説明](#)

[一般的な定義](#)

[コマンドおよび機能ロードマップ](#)

[Cisco IOS/Cisco IOS XE コールルーティングの基礎](#)

[音声ダイヤルピアのタイプ](#)

[着信ダイヤルピアの照合](#)

[一致するものが存在しない場合/Default Dial-Peer 0 peer_tag=0、pid:0](#)

[発信ダイヤルピアの照合](#)

[番号文字列ダイヤルピア ハンテイング](#)

[URI ダイヤルピア ハンテイング](#)

[voice class uri](#)

[着信 URI ダイヤルピアの照合](#)

[発信 URI ダイヤルピアの照合](#)

[ダイヤルピアのワイルドカード](#)

[ダイヤルピアの状態](#)

[Virtual Routing and Forwarding \(VRF\) とダイヤルピア ハンテイング](#)

[VRFによる着信ダイヤルピアの照合](#)

[VRFを使用した発信ダイヤルピアの照合](#)

[高度なコール ルーティング技術](#)

[ダイヤルピア グループ](#)

[E164-Pattern-Maps \(発信ダイヤラマップ\)](#)

[宛先サーバグループ](#)

[宛先サーバグループと OPTIONS キープアライブ](#)

[アウトバウンドプロキシ](#)

[POTS トランク グループ](#)

[音声クラス テナント](#)

[ILS URI コール CUBE \(音声クラス ルート文字列\)](#)

[従来のコール ルーティング技術](#)

[DNIS マップ](#)

[トランク グループ ラベル](#)

[番号タイプ](#)

[ダイヤルピア データ](#)

[音声ソースグループ](#)

[ダイヤルビアの権限](#)

URI とディジット操作

[POTS ダイヤルビアによるディジット操作](#)

[ボイストランスレーションルールとボイストランスレーションプロファイルによるディジット操作](#)

[音声クラスe164変換](#)

[ISDN マップによるディジット操作](#)

[番号拡張 \(num-exp \) によるディジット操作](#)

[着信/発信 SIP プロファイル](#)

[SIP Copylist](#)

特記事項

[プロトコルシグナリングとメディアバインド](#)

[DNS および VoIP ダイヤルビア](#)

[最大接続と帯域幅](#)

[ダイヤルイン方式 \(DID \)](#)

[1 段階ダイヤリング](#)

[2 段階ダイヤリング](#)

[コールのブロック](#)

[ISDN オーバーラップ受信](#)

[空の着信者番号](#)

[制限クラス](#)

[Cisco Unified Communications Manager Express \(CUCME \) ダイヤルビア](#)

[MGCP および SCCP とダイヤルビア](#)

[ダイヤルビアを使用したSIP DSAPP](#)

[コールルーティングのトラブルシューティングと確認](#)

はじめに

このドキュメントでは、Cisco IOS® および Cisco IOS XE コールルーティングについて説明します。

前提条件

要件

このドキュメントの読み取りに必要な正式な前提条件はありませんが、読者が、電話コールの確立と接続に使用される基盤となる音声シグナリングプロトコルについてある程度の知識を持っていることを前提に書かれています。これらのプロトコルは、全体を通じて何度も参照されます。

シグナリングプロトコル : Session Initiation Protocol(SIP)、H323(h225/h245)、メディアゲートウェイコントロールプロトコル(MGCP)、Skinny Client Control Protocol(SCCP)、ISDN Q931、E1 R2。

メディアプロトコル : リアルタイムプロトコル(RTP)、音声コーデック、ビデオコーデック

アナログテクノロジー : Ear and Mouth(E&M)、Foreign Exchange Subscriber(FXS)、Foreign

Exchange Office(FXO)。

使用するコンポーネント

このドキュメントの情報は、次のソフトウェアとハードウェアのバージョンに基づいています。

- Cisco IOSおよびIOS XEゲートウェイ
- 2800/3800/2900/3900/4300/4400/CSR1000v/CAT8000v/ASR100X/C8200/C8300/ISR1100

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、クリアな(デフォルト)設定で作業を開始しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

背景説明

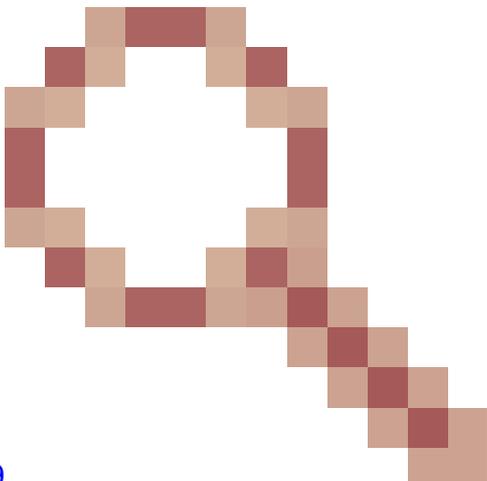
このドキュメントでは、一般電話サービス(POTS)およびVoice over IP(VoIP)ネットワークコールレグを使用した、着信および発信ダイヤルピア照会の背後にあるメカニズムについて説明します。

このドキュメントでは、ダイヤルピアの情報に加えて、コールルーティングに関連する重要なトピックについて説明します。これには、ディジット操作、Session Initiation Protocol (SIP) メッセージ操作の概要、通話機能を制限するいくつかの方法、メディアとシグナリングのバインドの概要、最後に若干のトラブルシューティングが含まれます。

このドキュメントでは、参照ポイントとして、設定例と、デバッグおよび show コマンドの出力を利用します。このドキュメントの多くの機能は、その機能がCisco IOSとCisco IOS XEの両方に導入されたバージョンによって明確に示されています。この情報は、「[コマンドと機能のロードマップ](#)」セクションでもすぐに参照できます。非常に重要な不具合がある場合は、テキスト内でリンクされているため、読者が認識できます。

一般的な定義

Attribute	説明
ディジット ストリング	番号文字列、電話番号、番号、またはE164番号とも呼ばれます。数字0 ~ 9のみで構成され、オプションで先頭にプラス記号(+)が付きます。 以下に例を挙げます。 8675309 123456789 +1972525222 +442084445555

Attribute	説明
	+85225353333
着信番号識別サービス (DNIS)	これは、通話の着信者番号または接続先番号です。
自動番号識別 (ANI)	これは、通話の発信者番号または送信側発信者番号です。これは、発信者IDというタイトルの発信者回線ID(CLID)とも呼ばれます。
Uniform Resource Identifier (URI)	<p>URIはsip : またはtel : のいずれかの文字列で、VoIPプロトコルSIPおよびH323で最も一般的に使用されます。</p> <p>URL例 :</p> <p>sip:user@host.com sip:user@sub.host.com sip:user@10.10.10.10 sip:user@2001:4860:4860::8888 tel:8675309 sip:host.com</p>
キャリア ID	<p>CID の例 :</p> <p>cid:orange@host.com cid:orange@sub.host.com cid:orange@10.10.10.10 cid:orange@2001:4860:4860::8888</p>  <p>注: Cisco Bug ID CSCua14749 Carrier-IDはIOS XEプラットフォームでは動作しません。</p>

Attribute	説明
ルート文字列	SIP で使用される ILS ルート文字列の Cisco 独自のヘッダー。 以下に例を挙げます。 X-cisco-dest-route-string: <sip : 設定値>
ENUM	ENUM は、ドメイン ネーム サービス (DNS) を使用して E164 電話番号を URI に変換するプロトコルです。これについての説明は、このドキュメントの対象外です。
PSTN	Public Switched Telephone Networks
ITSP	インターネットテレフォニー サービス プロバイダー
SBC	セッション ボーダー コントローラ。 これは、お客様の LAN と ITSP/PSTN ネットワーク間の境界ポイントとなるデバイスです

コマンドおよび機能ロードマップ

機能	IOS バージョン	IOS XE バージョン
番号拡張 (num-exp) ダイヤルピア (POTS および VoIP) answer-address destination-pattern incoming called-number session target (IPv4 および DNS) 最大接続数 (max-conn)	11.3(1)T	すべて

direct-inward-dial forward-digits (POTS) prefix (POTS) timeouts inter-digit (voice-port)		
dial-peer terminator	12.0	すべて
huntstop	12.0(5)T	すべて
ISDN マップ	12.0(6)T	すべて
ダイヤルピア ハンティング スキーム	12.0(7)XK	すべて
ボイス トランスレーショ ン ルールとボイス トラ ンスレーション プロファ イル translate-outgoing numbering-type digit-strip (POTS)	12.0.(7)XR1	すべて
session target (sip- server)	12.1(1)T	すべて
POTS トランク グループ	12.1(3)T	すべて
DNIS マップ (発信)	12.2(2)XB	すべて
trunk-group-label	12.2(11)T	すべて
ダイヤルピア (データ)	12.2(13)T	すべて

音声クラス URI (発信)	12.3(4)T	すべて
アウトバウンドプロキシ	12.4(15)T	すべて
session target (IPv6)	12.4(22)T	すべて
SIP プロファイル (発信)	15.0(1)M	すべて
音声クラス URI (着信) 音声ソースグループ	15.1(2)T	3.8S
SIP Copylist session target (レジストラ)	15.1(3)T	3.6S
call-route (url)	15.2(1)T	3.3S
max-bandwidth	15.2(2)T	3.7S
E164-Pattern-Maps (発信)	15.2(4)M	3.7S
音声クラス ルート文字列 call-route (dest-route-string)	15.3(3)M	3.10S
ダイヤルピア グループ (VoIP) E164-Pattern-Maps (着信) 宛先サーバ グループ requiri-passing session target (sip-uri)	15.4(1)T	3.11S

ダイヤルピア プロビジョン ポリシー SIP プロファイル (着信)	15.4(2)T	3.12S
ダイヤルピア グループ (POTS)	15.5(1)T	3.14S
音声クラス テナント	15.6(2)T	16.3.1
ダイヤルピアの VRF フィルタリング	15.6(3)M	16.3.1
e164 – 翻訳	該当なし	16.8.1
SIP DSAPP (必須)	該当なし	16.12.1
サーバグループのハンストップ	該当なし	17.4.1
ダイヤルピアのテナント フィルタリングのための SIPリスニングポート	該当なし	17.8.1
DNS SRVベースのオプションキープアライブ	該当なし	17.9.1

Cisco IOS/Cisco IOS XEコールルーティングの基礎

Cisco IOSおよびCisco IOS XEゲートウェイは、ダイヤルピアの概念を利用して、コールレグごとのコールルーティングと機能ネゴシエーションを制御します。コールレグは、2つのコールエージェント間の双方向通信です。コールエージェントは、テレフォニーコールを開始、処理、または転送するデバイスです。これは、テレフォニープロバイダー(TSP)機器、シスコゲートウェイ、IP電話、Cisco Unified Communication Manager(CUCM)、Cisco Unity Connection(CUC)などに限定されるものではありません。非常に多数のコールエージェントがあり、すべてを示すことはできません。

シナリオ：コールが別のコールエージェントからシスコゲートウェイに到達し、着信コールレグ(インレグ)になります。ゲートウェイはコールを処理し、その処理に基づいて次のコール

エージェントにコールを送信します。これは、発信コール レッグ (外部レッグ) です。

図 1 に、Cisco 音声ゲートウェイを介してルーティングされる PSTN から CUCM へのコールと、それぞれの着信および発信コール レッグ情報を示します。

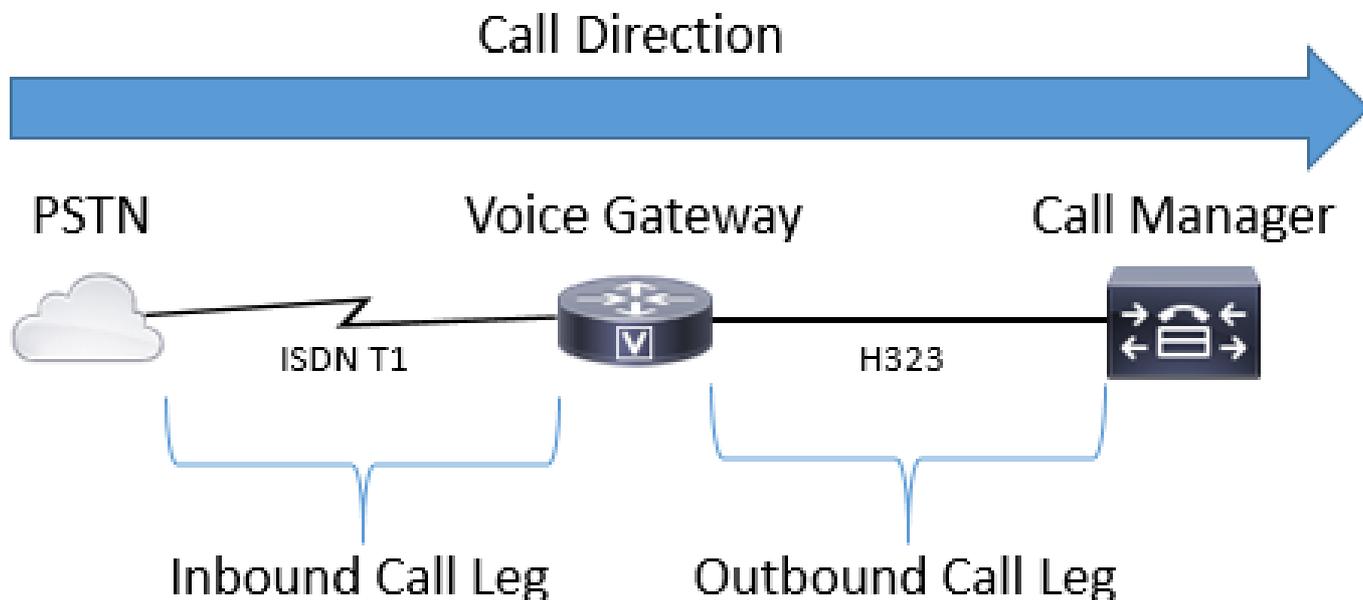


図 1 - 着信および発信コール レッグの図

Ciscoゲートウェイを経由する正常なコールは、着信または発信ダイヤルピアと常に一致し (注を参照)、適切にルーティングされます。着信および発信ダイヤルピアは、前述のコール レッグに似ています。図1では、コールはCiscoゲートウェイのPSTNから着信し、着信ダイヤルピアと照合する必要があります。ゲートウェイは発信ダイヤルピアを利用して、コールを次のコール エージェントにルーティングします。これらの用語は、Cisco ゲートウェイの観点から定義されていることに注意してください。

コールの各側のダイヤルピアを照合することにより、管理者は特定の各コールレッグのさまざまな側面を制御できます。たとえば、音声コーデック、DTMFプリファレンス、コールのルーティング先となるディジット操作、およびその他の多くの設定があります。ダイヤルピアは、着信と発信の両方の照合ステートメントで設定できるため、有効な着信および発信の照合設定が特定のダイヤルピアに適用された場合、内部レッグと外部レッグの両方で同じダイヤルピア照合が可能です。

 注：このルールの例外は、MGCPおよびSCCP音声ポートに関するものです。これらのシグナリング プロトコルは、コール ルーティング中に通常のダイヤルピア照合メカニズムに従いません。詳細については、「[SCCPおよびMGCP](#)」のセクションを参照してください。

図 2 は図 1 と同じ着信および発信コール レッグを示していますが、PSTN から CUCM へのコールのそれぞれのダイヤルピアは、Cisco 音声ゲートウェイ経由でルーティングされています。

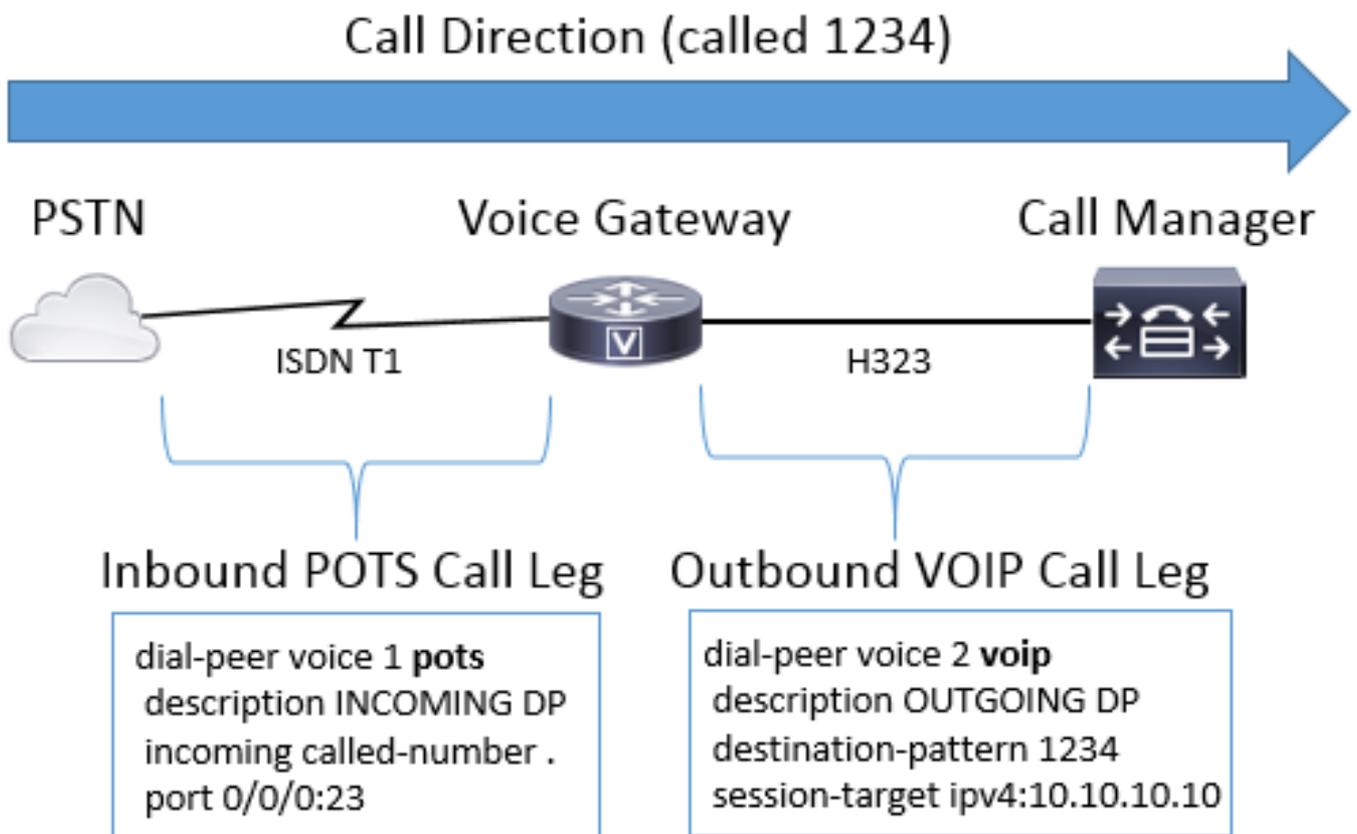


図 2 - 着信および発信ダイヤルピアの図

Cisco 音声ゲートウェイは、IP から IP、POTS から POTS、および IP から POTS またはその逆を含むさまざまな種類の音声コールとプロトコルを相互接続できます。

図 3 に、Cisco Unified Border Element (CUBE) を介した VoIP から VoIP へのコールを示します。

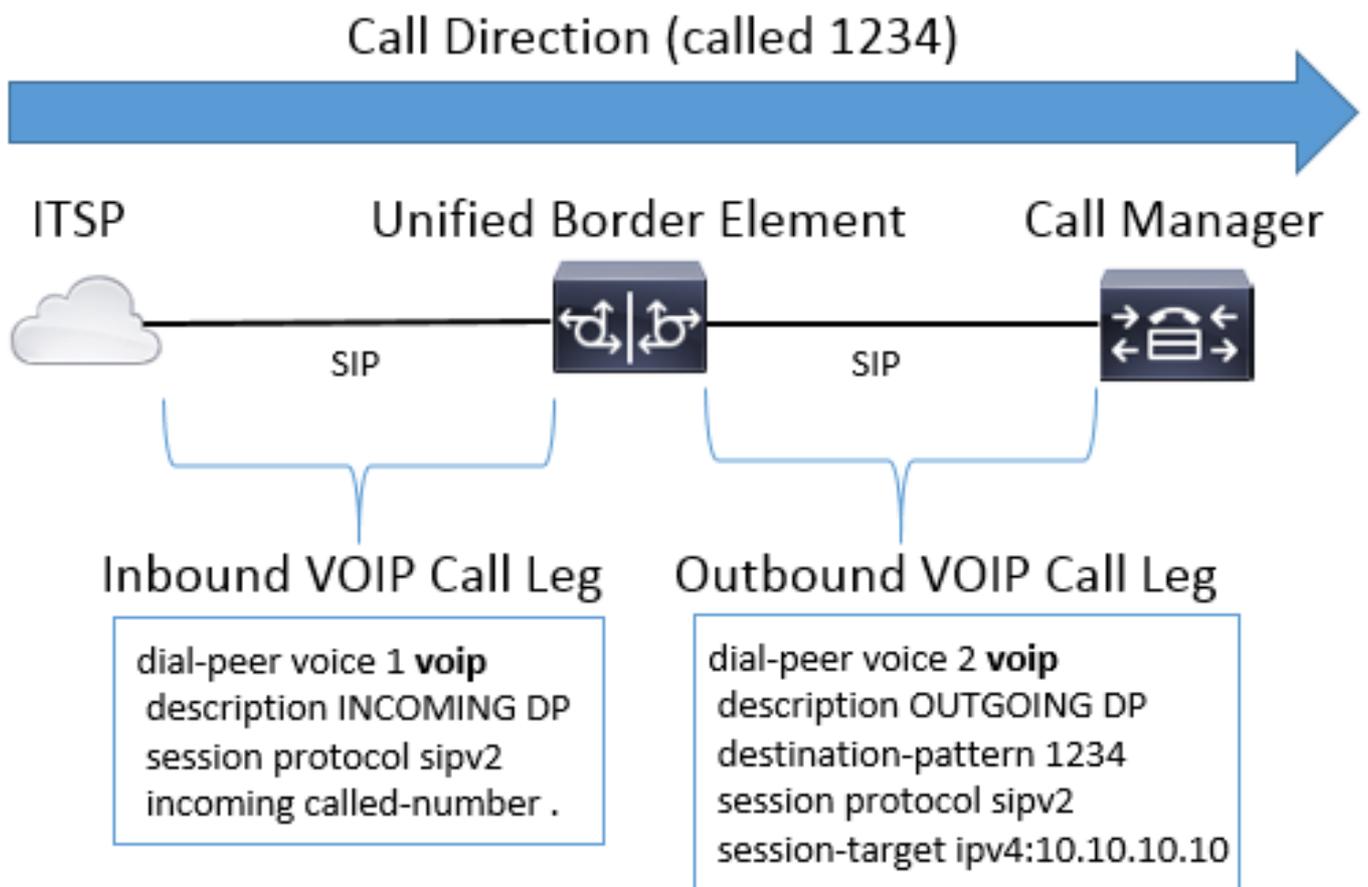


図 3 - VoIP から VoIP へのコールの着信および発信ダイヤルピア

図 4 に、Cisco ゲートウェイを介した POTS から POTS へのコールを示します。

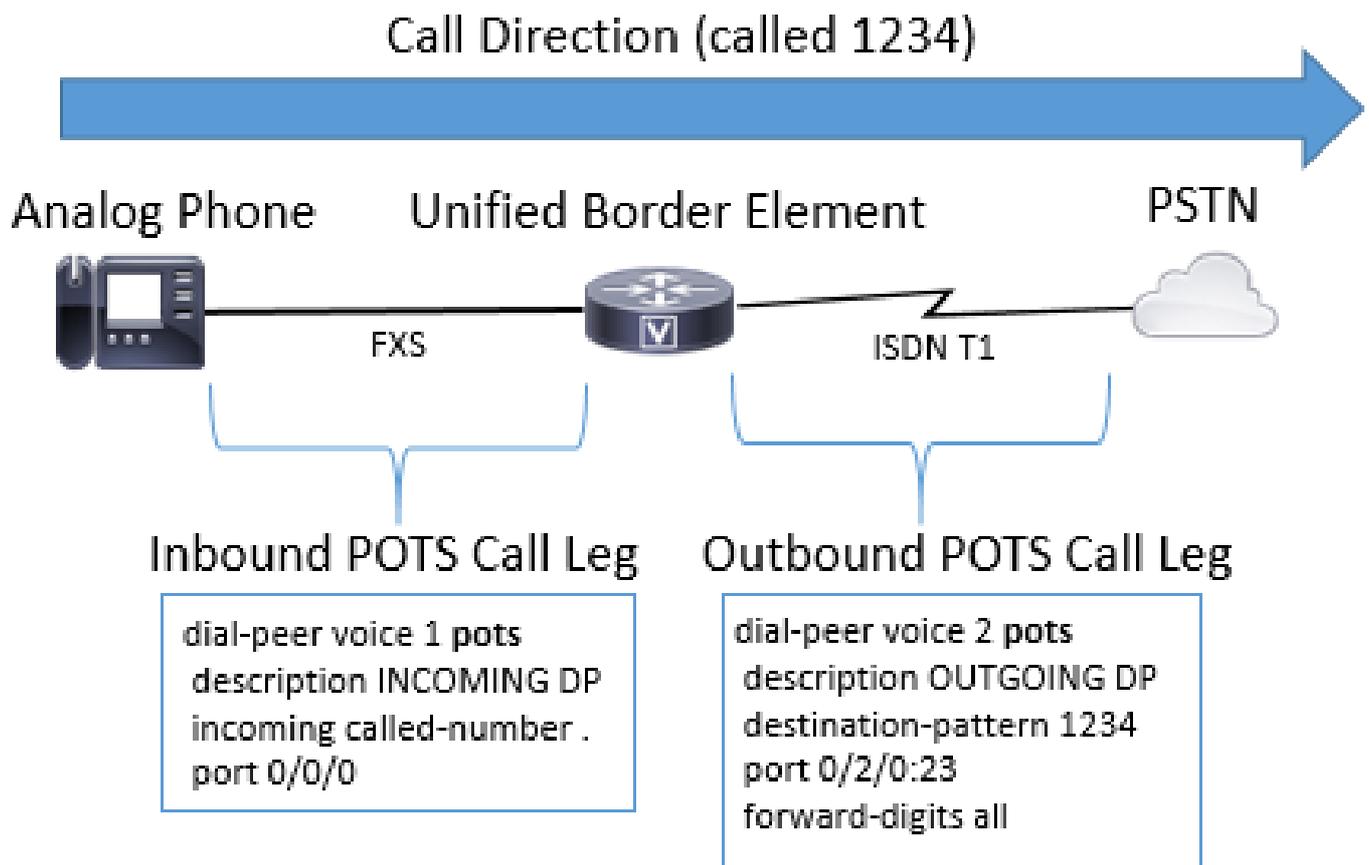


図 4 - POTS から POTS へのコールの着信および発信ダイヤルピア

音声ダイヤルピアのタイプ

POTS# POTS#	<p>単純な旧式のテレフォニーサービス(POTS)ダイヤルピアは、アナログFXS、FXO、ISDN T1/E1、E1 R2、Ear and Mouth(E&M)接続などのアナログ接続と照合されます。</p> <p>これらはゲートウェイの物理音声ポートとの間でコールを送受信します。</p>
VOIP	<p>Voice Over IP(VoIP)ダイヤルピアは、主にゲートウェイとの間のH323およびSIP接続を制御するために使用されます。</p> <p>これらのダイヤルピアは、ドメインネームシステム(DNS)を使用して、IPv4アドレスとIPv6アドレスの両方、および完全修飾ドメイン名(FQDN)からシグナリングを送受信します。</p> <p>—</p> <p>VoIP ダイヤルピアは、Voice over Frame Relay (VoFR)、Voice over ATM (VoATM)、Voice over High-Level Data Link Control (VoHDLC)、および Registration, Admission, and Status (RAS) のシグナリングにも使用するこ</p>

	<p>とができ、これらのダイヤルピアのセッション ターゲットには、解決と ENUM の値を含めることもできます。</p> <p>注：これらのタイプの設定の一部は、新しいネットワークでは見られない古いテクノロジーであり、IOS XEではサポートされなくなっているものもあります。そのため、それらの設定はこのドキュメントで取り上げていません。</p>
MMOIP	<p>Multimedia Mail over IP ダイヤルピアは、Exchange サーバに電子メールを送信するために使用します。</p> <p>これらは t37 オンランプ/オフランプ ファクスに主に使用されます。これらのダイヤルピア タイプについては、このマニュアルでは扱いません。</p>

 注: Cisco ゲートウェイで設定できるダイヤルピアの最大数は、使用可能なメモリ (DRAM) によって異なります。各ダイヤルピアは約 6KB のメモリを消費するため、ゲートウェイで他の CPU プロセス用に合計メモリの少なくとも 20% が予約されていることを確認してください。多数のダイヤルピアが設定されていると、コールをルーティングするための遅延が追加される可能性があります。Cisco 音声アプリケーションは、アクセス コントロール リスト (ACL) と同様にダイヤルピアをトップダウン方式で調べるため、これは大きな影響を与える場合があります。これは通常、新しい Cisco ゲートウェイでは問題になりません。

サンプルエラー：

```
May 26 12:59:46.406: %DIALPEER_DB-3-ADDPEER_MEM_THRESHOLD: Addition of dial-peers limited by available
```

着信ダイヤルピアの照合

Cisco ゲートウェイは、コール設定要求を受信すると、このコールに適用可能な着信ダイヤルピアの検索を開始します。これはデジタル単位の分析ではなく、メッセージ全体を使用してどの着信ダイヤルピアが選択されているかを判断します。メッセージでチェックされる項目の順序は、表 1、表 2、および表 3 で定義されている優先順位リストに示されるコールのプロトコルに大きく依存します。ダイヤルピアが満たす必要がある照合の条件は 1 つのみです。ダイヤルピアには必ずしもすべての属性を設定する必要はありません。また、すべての属性がコールセットアップ情報に一致する必要もありません。すべてのダイヤルピアは、最初の一致基準に基づいて検索されます。ゲートウェイは、一致が見つからない場合にのみ、次の基準に進みます。

表 1. 着信 SIP ダイヤルピアの選択プリファレンス

プリファレンス	一致基準	ダイヤルピアコマンド
---------	------	------------

1	URI	incoming uri via <uri-tag>
2	URI	incoming uri request <uri-tag>
3	URI	incoming uri to <uri-tag>
4	URI	incoming uri from <uri-tag>
5	Called Number	incoming called-number <number-string> incoming called e164-pattern-map <pattern-map-number>
6	Calling Number	incoming calling e164-pattern-map <pattern-map-number> answer-address <number-string>
7	宛先パターン (ANI)	destination-pattern <number-string>
8	キャリア ID	carrier-id source <string>

注：適格な着信ダイヤルピアは、VRFまたはテナントでフィルタリングできます（該当する機能が設定されている場合）。詳細については、「仮想ルーティングおよび転送 (VRF)」と「ダイヤルピアハンテイング」および「音声クラステナント」のセクションを参照してください。

表 2 着信H323ダイヤルピアの選択プリファレンス

プリファレンス	一致基準	ダイヤルピアコマンド
1	URI	incoming uri called <uri-tag> incoming uri calling <uri-tag>
2	Called Number	incoming called-number <number-string>

		incoming called e164-pattern-map <pattern-map-number>
3	Calling Number	incoming calling e164-pattern-map <pattern-map-number> answer-address <number-string>
4	宛先パターン (ANI)	destination-pattern <number-string>
5	キャリア ID	carrier-id source <string>

表 3 着信ブロックPOTSダイヤルピアの選択プリファレンス

プリファレンス	一致基準	ダイヤルピアコマンド
1	Called Number	incoming called-number <number-string>
2	Calling Number	answer-address <number-string>
3	宛先パターン (ANI)	destination-pattern <number-string>
4	音声ポート	port <voice-port-number>

一致が存在しない場合/デフォルトのダイヤルピア 0 (peer_tag=0、pid:0)

POTSまたはVoIPコールの着信ダイヤルピアに一致する有効な一致がない場合、ゲートウェイはダイヤルピア0を割り当てます。ダイヤルピア 0 は機能に制限があり、コールに問題を引き起こす可能性があるため、これは理想的ではありません。これに対する例外は、コールのルーティングにダイヤルピアを使用しないSCCPおよびMGCPプロトコルです。詳細については、[MGCP と SCCP の項を参照してください。](#)

ダイヤルピア 0 の機能

- dtmf-relayメカニズムはありません。
- VoIPコールのすべての音声コーデックをアダプタイズしました。
- Fax-rate voice.
- Voice Activity Detection (VAD ; 音声アクティビティ検出) がイネーブルになっている。
- no rsvp support.
- POTSコールに対するIVRアプリケーションサポートはありません。

- Direct-inward-dialが有効である。
- VRFはサポートされません。

発信ダイヤルピアの照合

発信ダイヤルピアは、POTS または VoIP コールをゲートウェイから次のコール エージェントにルーティングするために使用されます。着信ダイヤルピア照合と同様に、特定のプロトコルの優先順位に基づいてダイヤルピアを照合するためにゲートウェイが使用できる項目のリストがあります。ただし、着信ダイヤルピアとは異なり、コールをルーティングする適切な発信ダイヤルピアがない場合、コールは失敗します。着信ダイヤルピア照合と同様に、すべてのダイヤルピアは最初の一貫基準に基づいて検索されます。ゲートウェイは、一致が見つからない場合にのみ、次の基準に進みます。

表 4 発信SIPダイヤルピアの選択プリファレンス

プリファレンス	一致基準	ダイヤルピアコマンド
1	ダイヤルピア グループのダイヤルピア	destination dpg <dpg-tag> (着信ダイヤルピアで設定されたDPG)
2	ダイヤルピア プロビジョン ポリシー URI	destination uri-from <uri-tag> destination uri-to <uri-tag> destination uri-via <uri-tag> destination uri-diversion <uri-tag> destination uri-referred-by <uri-tag> (着信ダイヤルピアで設定されたDPP)
3	ILS ルート文字列	destination route-string <route-string-tag>
4	URI とキャリア ID	destination uri <uri-tag> および carrier-id target <string>
5	着信者番号とキャリア ID	destination-pattern <number-string> および carrier-id target <string>
6	URI	destination uri <uri-tag>
7	Called Number	destination-pattern <DNIS-number>

		destination e164-pattern-map <pattern-map-number> dnis-map <dnis-map-number>
8	Calling Number	destination calling e164-pattern-map <pattern-map-number>

表 5 発信H323ダイヤルピアの選択プリファレンス

プリファレンス	一致基準	ダイヤルピアコマンド
1	ダイヤルピア グループのダイヤルピア	destination dpg <dpg-tag> (着信ダイヤルピアで設定)
2	URI とキャリア ID	destination uri <uri-tag> および carrier-id target <string>
3	着信者番号とキャリア ID	destination-pattern <number-string> および carrier-id target <string>
4	URI	destination uri <uri-tag>
5	Called Number	destination-pattern <number-string> destination e164-pattern-map <pattern-map-number> dnis-map <dnis-map-number>
6	Calling Number	destination calling e164-pattern-map <pattern-map-number>

表 6 発信POTSダイヤルピアの選択プリファレンス

プリファレンス	一致基準	ダイヤルピア コマンド*

1	ダイヤルピア グループのダイヤルピア	destination dpg <dpg-tag> (着信ダイヤルピアで設定)
2	URI とキャリア ID	destination uri <uri-tag> および carrier-id target <string>
3	着信者番号とキャリア ID	destination-pattern <number-string> および carrier-id target <string>
4	URI	destination uri <uri-tag>
5	Called Number	destination-pattern <DNIS-number>dnis-map <map-number>

 注: 「[番号文字列ダイヤルピアハンティング](#)」および「[URIダイヤルピアハンティング](#)」のセクションでは、ゲートウェイが次の一致基準に移動する前に各一致基準行の潜在的なコマンドのリストを評価する方法について説明しています。たとえば、すべての潜在的な destination-pattern matching コマンドと destination e164-pattern-map matching コマンドを評価してから、calling number コマンドを調べます。

番号文字列ダイヤルピア ハンティング

番号文字列のプリファレンス :

URI がマッチを評価するための特定の操作順序を持っているのと同様に、数字のディジットストリングを評価する際に使用される一連のルールもあります。Cisco ゲートウェイのデフォルトのダイヤルピアハントスキームは 0 に設定されています。このことは、ゲートウェイが最長一致 (最も固有) のパターンを検索することを意味します。一致長が同じダイヤルピアが 2 つある場合、ゲートウェイは明示的に定義されたダイヤルピアプリファレンスを参照します。最後に、両方が同じ場合は、ランダムな順序で選択されます。

設定に使用できる他のダイヤルピアハントスキームがありますが、ほとんどの導入ではデフォルトの 0 のままです。

 ヒント : ダイヤルピアがデフォルトの順序以外で照合される場合、管理者はデフォルト以外のダイヤルピアハントスキームの実行コンフィギュレーションを調べることができます。

```
Gateway(config)# dial-peer hunt ?
<0-7> Dial-peer hunting choices, listed in hunting order within each choice:
0 - Longest match in phone number, explicit preference, random selection.
```

- 1 - Longest match in phone number, explicit preference, least recent use.
- 2 - Explicit preference, longest match in phone number, random selection.
- 3 - Explicit preference, longest match in phone number, least recent use.
- 4 - Least recent use, longest match in phone number, explicit preference.
- 5 - Least recent use, explicit preference, longest match in phone number.
- 6 - Random selection.
- 7 - Least recent use.

最長一致番号文字列ダイヤルピアアルゴリズムは、番号文字列内の番号のシーケンスと完全に一致するシーケンス内の番号が最も多いダイヤルピアを見つけます。この概念は、次のシナリオで明確になります。

シナリオ：条件を満たすダイヤルピアにこれらの一致が設定されており、ゲートウェイは2001のディジットストリングを評価しています。ダイヤルピア1は2000～2999のいずれかの数字と一致する可能性があり、ダイヤルピア2は2000～2009と一致する可能性があります。ダイヤルピア2はディジットストリング2001の最長一致（最も固有）であるため、デフォルトのダイヤルピアハンティングメカニズムが採用された場合（ダイヤルピアハント0）、このコールに一致します。つまり、番号のシーケンス200は、番号文字列2001の番号のシーケンスと完全に一致する最大のシーケンスです。

```
!  
dial-peer voice 1 voip  
  destination-pattern 2...  
!  
dial-peer voice 2 voip  
  destination-pattern 200.  
!
```

プリファレンスは、各ダイヤルピアの管理者定義のウェイトとして定義されます。管理者は、コールが常に他のダイヤルピアよりも先に特定のダイヤルピアを使用するようにプリファレンスを設定できます。デフォルトでは、すべてのダイヤルピアがプリファレンス0です。プリファレンス0のダイヤルピアは、プリファレンス1～10の他のダイヤルピアの前に照合されます。ほとんどの管理者は、バックアップサブスクライバがある特定のCUCMサブスクライバ、または優先度の低い（高い番号で設定された）別のダイヤルピアを使用して別のコールエージェントが設定されている特定のCUCMサブスクライバにコールを送信するように複数のダイヤルピアを設定します。

シナリオ：2つのダイヤルピアが、ディジットストリング2001に対して同じ一致長で設定されています。管理者は明示的なプリファレンスを定義します。一致長が同じであるため、ゲートウェイは両方のダイヤルピアを同じと評価します。ただし、管理者はダイヤルピア1のプリファレンスをより高く設定し、ダイヤルピア1がコールのルーティングで使用される最初のダイヤルピアとして選択されるようにします。最初のダイヤルピアで障害が発生する可能性があるため、ダイヤルピア2はセカンダリオプションのままになります。

```
!  
dial-peer voice 1 voip
```

```
destination-pattern 2...
preference 1
!
dial-peer voice 2 voip
destination-pattern 2...
preference 2
!
```

Ciscoゲートウェイは、一度に1つの適切な発信ダイヤルピア経由でのみコールのルーティングを試みます。最初に選択したダイヤルピアで障害状態が見つかった場合、ゲートウェイは次の適切なダイヤルピアでコールのルーティングを試行します。これは、コールが成功するか、試行する適切なダイヤルピアが残っていないために失敗するまで続行されます。ダイヤルピアハンティングと障害の一般的な症状として、発信時の呼び出し音の大幅な遅延があります。通常、デバッグは、特定のダイヤルピアでコールが失敗する理由を正確に確認するために必要です。障害状態が見つかった際に、管理者がゲートウェイで他のダイヤルピアを検索しない場合は、ダイヤルピアでコマンド `huntstop` を使用できます。

シナリオ:2つのダイヤルピアが、ディジットストリング2001に対して同じ一致長で設定されています。管理者は明示的なプリファレンスを定義しており、この特定のコールに対してダイヤルピア2を照合しません。一致長が同じダイヤルピアが2つあるため、ダイヤルピアの決定にはプリファレンスが使用されます。ダイヤルピア1には最も低い設定番号が割り当てられているため、この番号を使用してコールがルーティングされます。dial-peer 1を使用している発信コールレグで障害状態が発生すると、`huntstop`コマンドが設定されているため、ゲートウェイは直ちにダイヤルピアハンティングを停止します。このシナリオでは、ダイヤルピア2が発信ルーティングに使用されることはありません。

```
!
dial-peer voice 1 voip
destination-pattern 2...
preference 1
huntstop
!
dial-peer voice 2 voip
destination-pattern 2...
preference 2
!
```

 注：一般的なダイヤルピア設定コマンドであるため、`huntstop`コマンドと`preference`コマンドをURI照合ステートメントと組み合わせて使用することもできます。さらに、音声クラスのサーバグループ設定では、17.4.1aで`huntstop`コマンドを使用できます。詳細については、「宛先サーバグループ」のセクションを参照してください。

URI ダイヤルピア ハンティング

ゲートウェイは各一致基準を調べ、その一致基準を使い果たしてから、次の一致基準に進みます。たとえば、着信SIPコールに対して行います。[表1](#)に基づきます。[着信SIPダイヤルピア選択プ](#)

リファレンスでは、Ciscoゲートウェイが最初にチェックするのはURIであり、すべての潜在的なURIコマンドを評価して、適合するURIを見つけます。一致がないか、または設定されていない場合、ゲートウェイは次の一致項目に移動し、その基準に基づいて評価を実行します。このプロセスは、一致に基づいてコールがルーティングされるか、またはゲートウェイがチェックする一致基準がなくなるまで繰り返されます。

着信または発信ダイヤルピアがURIコマンドで設定されると、ゲートウェイは複数のヘッダーで受信されたURIを調べて、一致する可能性があるかどうかを確認します。一致プリファレンスは最も固有の一致に基づき、正確なプリファレンスは、URI全体の一致、ホスト部分、ユーザ部分、または電話URIの順になります。URI照合の操作順序を知ると、SIPおよびCUBE展開でのダイヤルピア照合に非常に役立ちます。

このプリファレンスの順序は、コマンド `voice class uri sip preference` を使用して操作して、ホストではなくユーザIDを最初のオプションとして指定できます。

URIのプリファレンス：

1. URIのホスト部分。例：(`@a.b.c.d`または`@host.domain.name`)
2. URIのユーザ部分。例：(`sip:8675309`または`sip:user`)
3. 電話番号URI。例：(`tel:18005532447`)
4. URI全体の完全一致。例：(`user@host.domain.name`、`user@a.b.c.d`、`8675309@host.domain.name`、`8675309@host.domain.name`)

サポートドキュメント：[Cisco Unified Border Elementコンフィギュレーションガイド - Cisco IOS XE 17.6以降](#)

シナリオ：管理者がこのダイヤルピアを設定し、ゲートウェイにコールを送信しました。受信したInviteのFromヘッダーは、From: `<sip:testuser@10.10.10.10>`です。ゲートウェイは、このヘッダーに基づいて、2つの異なるダイヤルピアを潜在的に照合できます。ユーザ部分に基づくダイヤルピア1と、ホスト部分に基づくダイヤルピア2です。ただし、ホストの一致はユーザの一致よりも優先されるため、ダイヤルピア2がコールの着信ダイヤルピアに使用されます。

```
!  
voice class uri URI1 sip  
  user-id testuser  
!  
voice class uri URI2 sip  
  host ipv4:10.10.10.10  
!  
dial-peer voice 1 voip  
  sess protocol sip2  
  incoming uri FROM URI1  
!  
dial-peer voice 2 voip  
  sess protocol sip2  
  incoming uri FROM URI2  
!
```

voice class uri

着信および発信ダイヤルピアの URI 照合では、管理者はメッセージングで URI をサポートする VoIP プロトコルの電話番号文字列より多くの照合を柔軟に実行できます。IOS 15.4(1)T および IOS-XE 3.11S よりも前のリリースでは、要求 URI に英数字の user@host を含める必要があり、含めない場合は、シスコゲートウェイが 4xx メッセージでコールを拒否します。これで、URI にホスト部分だけを含めることができ、ゲートウェイは提供されたホストだけに基づいてコールをルーティングします。たとえば、sip:cisco.com などです。

また、IOS 15.4(1)T および IOS-XE 3.11S 以前は、voice-class URI ユーザ ID は数字の E.164 値 (sip:1234@host.com) のみが可能でした。これは、管理者が CUBE(sip:user@host.com) で英数字のユーザ ID を設定できるように変更されました。

voice class uri のホスト部分またはユーザ部分には、正規表現(regex)パターンを含めることができ、一致する可能性のある値が大幅に増加します。

```
Gateway(config-voice-uri-class)# user-id .)
% unmatched ()user-id pattern can be of format ^([0-9A-Za-z\|\/\(\)*+^$&?#--.])*$
```

```
Gateway(config-voice-uri-class)# host .)
% unmatched ()host pattern can be of format ^([0-9A-Za-z\|@\/\(\)*+^$&?#--.])*$
```

```
Gateway(config-voice-uri-class)# pattern .)
% unmatched ()pattern pattern can be of format ^([0-9A-Za-z\|@;:=%!\~\/\(\)*+^$&?#--.])*$
```

例：音声クラスURI

```
!
voice class uri HOST sip
  host webex.com
  host dns:cisco.webex.com
  host ipv4:10.50.244.2
  host ipv6:[2001:4860:4860::8888]
!
voice class uri USER sip
  user-id username
!
voice class uri PATTERN sip
  pattern 8675309
!
voice class uri HostRegex sip
  host (.*)cisco.com
!
voice class uri ipRegex sip
  host 172\.18\.110\.20[567]
!
voice class uri PatternRegex sip
  pattern 555(.*)
!
voice class uri ipRegex sip
  pattern (172\.18\.110\.10[134]|10\.10\.10\.10)
```

```
! One Line that matches 172.18.110.101, 172.18.110.103, 172.18.110.104 OR 10.10.10.10
!  
voice class uri UserRegex sip  
  user-id test(.*)  
!
```

この例で示すように、voice class uriごとに設定できるホストは10台だけ、パターンは1つ、ユーザIDは1つです。一致する必要がある項目が多い場合は、正規表現を使用することをお勧めします。

```
Gateway(config)# voice class uri TEST sip  
Gateway(config-voice-uri-class)#host ipv4:10.1.1.1  
Gateway(config-voice-uri-class)#host ipv4:10.2.2.2  
Gateway(config-voice-uri-class)#host ipv4:10.3.3.3  
Gateway(config-voice-uri-class)#host ipv4:10.4.4.4  
Gateway(config-voice-uri-class)#host ipv4:10.5.5.5  
Gateway(config-voice-uri-class)#host ipv4:10.6.6.6  
Gateway(config-voice-uri-class)#host ipv4:10.7.7.7  
Gateway(config-voice-uri-class)#host ipv4:10.8.8.8  
Gateway(config-voice-uri-class)#host ipv4:10.9.9.9  
Gateway(config-voice-uri-class)#host ipv4:10.10.10.10  
Gateway(config-voice-uri-class)#host ipv4:10.11.11.11  
Error:Maximum of 10 hosts can only be configured.
```

```
Gateway(config)# voice class uri TEST2 sip  
Gateway(config-voice-uri-class)#host dns:1.com  
Gateway(config-voice-uri-class)#host dns:2.com  
Gateway(config-voice-uri-class)#host dns:3.com  
Gateway(config-voice-uri-class)#host dns:4.com  
Gateway(config-voice-uri-class)#host dns:5.com  
Gateway(config-voice-uri-class)#host dns:6.com  
Gateway(config-voice-uri-class)#host dns:7.com  
Gateway(config-voice-uri-class)#host dns:8.com  
Gateway(config-voice-uri-class)#host dns:9.com  
Gateway(config-voice-uri-class)#host dns:10.com  
Gateway(config-voice-uri-class)#host dns:11.com  
Error:Maximum of 10 hosts can only be configured.
```

```
Gateway(config)# voice class uri TEST3 sip  
Gateway(config-voice-uri-class)#user-id 8675309  
Gateway(config-voice-uri-class)#user-id 123456789  
Gateway(config-voice-uri-class)#do sh run | s TEST3  
voice class uri TEST3 sip  
user-id 123456789
```

```
Gateway(config)# voice class uri TEST4 sip  
Gateway(config-voice-uri-class)#pattern 8675309  
Gateway(config-voice-uri-class)#pattern 123456789  
Gateway(config-voice-uri-class)#do sh run | s TEST4  
voice class uri TEST4 sip  
pattern 123456789
```

着信 URI ダイアルピアの照合

この機能は IOS 15.1(2)T と IOS-XE 3.8S で追加され、着信ダイヤルピアに設定され適用された voice class uri を利用します。着信URIは、着信ダイヤルピアの選択時にチェックされる最初の一致基準であるため、多くのユーザでSIPコールに対して従来のincoming called-numberステートメントよりも採用されています。このコマンドを使用すると、管理者は、特定のコールエージェントまたはユーザからのコールをよりの確に照合することもできます。

詳細なドキュメント：[Cisco Unified Border Elementコンフィギュレーションガイド - Cisco IOS XE 17.6以降](#)

一般的な使用例

1. CUCM からの OPTIONS ping 要求に応答するための、URI のホスト部分の着信ダイヤルピア照合。
2. インターネット テレフォニー サービス プロバイダー (ITSP) からの着信コールを制御するための、URI のホスト部分の着信ダイヤルピア照合。
3. 特定のユーザまたは番号からのコール処理のためにURIのユーザID部分で照合される着信ダイヤルピア。

設定例

この出力例では、音声クラスURIで定義された2つのホストIPから送信されるすべてのSIP要求で、ダイヤルピア777が一致します。監視されるヘッダーは、ダイヤルピアのFromヘッダーとして定義されます。ただし、管理者はVIA、TO、REQUEST (要求URI) など、その他の多くのヘッダーを定義できます。CUCMがCUBEにOPTIONS pingを送信する場合、ダイヤルピア777を照合し、200 OKの応答を指定されたインターフェイスからOPTIONSに送信します。CUCMがCUBEにInviteを送信する場合、ダイヤルピア777を着信ダイヤルピアとして照合します。

```
!  
voice class uri CUCM sip  
  host ipv4:10.50.244.2  
  host ipv4:10.50.244.20  
!  
dial-peer voice 777 voip  
  description INCOMING URI  
  session protocol sipv2  
  incoming uri from CUCM  
  voice-class sip bind control source-interface Loopback777  
  voice-class sip bind media source-interface Loopback777  
!
```

発信 URI ダイヤルピアの照合

Cisco IOSゲートウェイは、voice class uri を発信ダイヤルピアに適用し、call-route urlをグローバル設定に追加することによって、URIを使用して発信ダイヤルピアを照合できます。これが存在する場合、CUBEは要求URIに基づいてコールのルーティングを試行できます。この機能はIOS 12.3(4)Tで追加され、すべてのIOS XEバージョンに存在します。デフォルトでは、発信SIP要求URIとToヘッダーURIに発信ダイヤルピアのセッションターゲットがあることに注意してください

い。これは、コマンド `requiri-passing` を使用して無効にすることができます。これにより、ゲートウェイは URI のホスト部分をセッション ターゲットに置き換える代わりに、内部レグ URI のホスト部分を外部レグに渡すことができます。コマンド `requiri-passing` は、15.4(1)TとIOS XE 3.11Sで追加されました。

設定例

```
voice service voip
  sip
    call-route url
    requiri-passing
!
voice class uri CUCM sip
  host dns:.*.com
!
dial-peer voice 777 voip
  description OUTGOING URI
  session protocol sipv2
  destination uri CUCM
  session target sip-uri
!
```

出典：[Cisco Unified Border Elementコンフィギュレーションガイド – Cisco IOS XE 17.6以降](#)

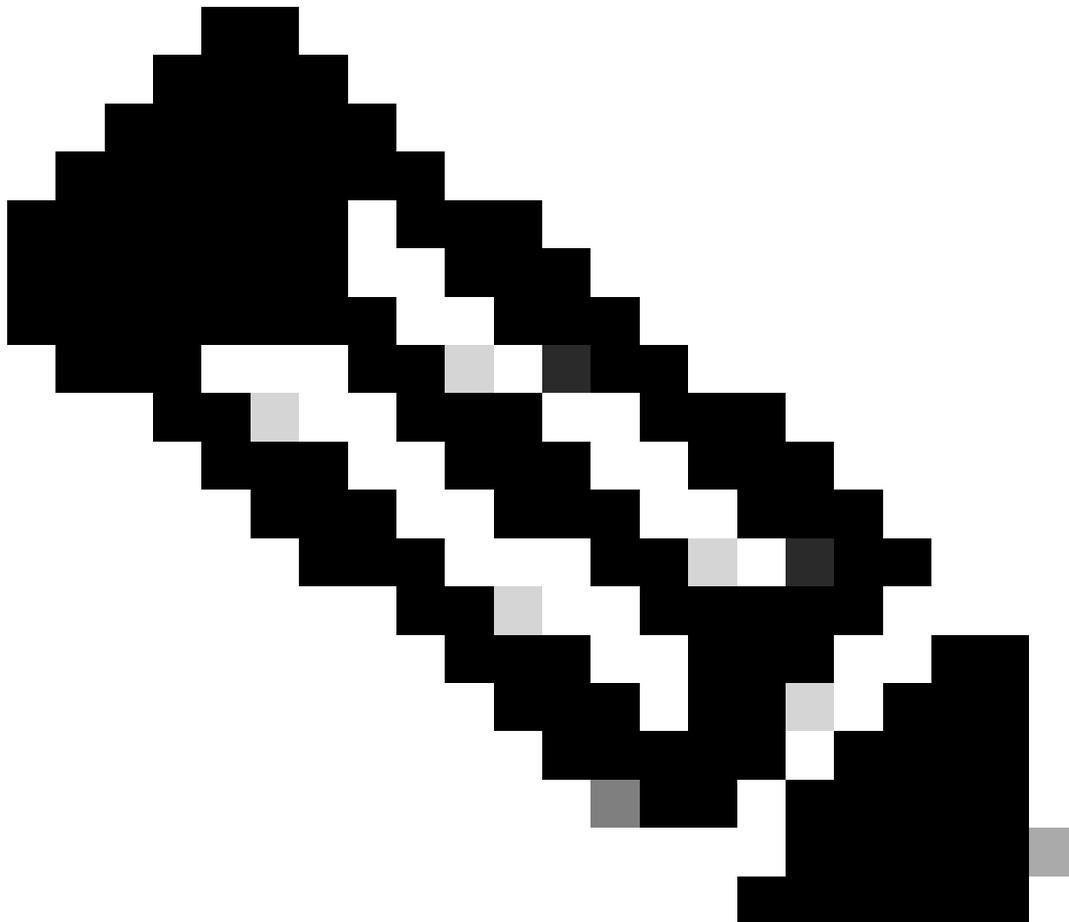
管理者は、音声クラスURIに加えて、`dial-peer provision-policy(DPP)`を使用して、発信ダイヤルピア照合のインレグURIを照合できます。この機能は、IOS 15.4(2)TおよびIOS XE 3.12Sで追加されました。`dial-peer provision-policy` では、セカンダリー一致属性をオプションとしてプライマリー一致属性を定義する必要があります。`provision-policy`は着信ダイヤルピアに適用され、そのダイヤルピアが着信コールレグで使用するために選択されると、ポリシーが起動されます。その結果、`dial-peer provision-policy` の属性に基づいて発信ダイヤルピアが選択されます。

発信照合は、単一のヘッダーでも複数のヘッダーでも可能です。ダイヤルピアを照合するにはこれらすべてがtrueである必要があります。

この例では、FromヘッダーとToヘッダーに音声クラスuriがあります。OR照合では、2つのプリファレンスを含む`dial-peer provision-policy`が設定されます。Fromヘッダーが最初のプリファレンスで、Toヘッダーがバックアップのプリファレンスです。ダイヤルピア1234は、着信照合にプロビジョニングポリシーを適用するように構築されています。次に、`destination uri-from`コマンドと`destination uri-to`コマンドをそれぞれ適用するダイヤルピア11111と22222を構築します。これらのコマンドは、それぞれの音声クラスURIを指します。コールに対して、Invite、matchダイヤルピア1234を受信し、`provision-policy`を確認できます。その後、デバイスは、ダイヤルピア11111-トで適用可能な一致として、最初にFromヘッダーでのルーティングを試行できます。これが失敗した場合は、22222を使用してtoヘッダーでのルーティングを試みることもできます。

また、ダイヤルピアの`provision-policy`で「And」照合を行う方法の詳細も示します。同じInviteが受信されると仮定して、1つのプリファレンスの下に2つのヘッダーを定義し、これを着信ダイヤルピアに適用できます。

これでINVITEを受信したときに、provision-policyで定義されている両方の一致基準を満たす適格な発信ダイヤルピアをチェックできるようになりました。したがって、この例では、照合するために発信ダイヤルピアはTOヘッダーとFROMヘッダーの両方で定義される必要があります。いずれかが有効な一致ではない場合、このダイヤルピア12345は使用されません。



注：Fromヘッダーでコールをルーティングしても、ゲートウェイから発信されるInviteには元の要求URIが残っています。ダイヤルピアのprovision-policyを使用して、発信ダイヤルピアを照合し、要求URIは変更しません。

設定例：

<#root>

Received INVITE

Received:
INVITE sip:8675309@172.18.110.58:5060 SIP/2.0
From: sipp <sip:sipp@172.18.110.65>;tag=1
To: sut <sip:cube@172.18.110.58:5060>

Common Configurations

```
!  
voice class uri FROM sip  
  user-id sipp  
!  
voice class uri T0 sip  
  user-id cube  
!
```

OR Match

```
!  
voice class dial-peer provision-policy 1  
  description match from header. If false, try to header  
  preference 1 from  
  preference 2 to  
!  
dial-peer voice 1234 voip  
  session protocol sipv2  
  destination provision-policy 1  
  incoming called-number .  
!  
dial-peer voice 11111 voip  
  destination uri-from FROM  
  session protocol sipv2  
  session target ipv4:172.18.110.48  
!  
dial-peer voice 22222 voip  
  destination uri-to T0  
  session protocol sipv2  
  session target ipv4:172.18.110.48  
!
```

AND Match

```
!  
voice class dial-peer provision-policy 2  
  description match from AND to headers  
  preference 1 from to  
!  
dial-peer voice 1234 voip  
  session protocol sipv2  
  destination provision-policy 2  
  incoming called-number .  
!  
dial-peer voice 12345 voip  
  destination uri-from FROM  
  destination uri-to T0  
  session protocol sipv2  
  session target ipv4:172.18.110.48  
!
```

session target sip-uri

IOS 15.4(1)TおよびIOS XE 3.11Sよりも前のリリースでは、URIのホスト部分が異なってもユーザが同じ場合、2つの別個の発信ダイヤルピアが必要でした。

このリリース以降、管理者は1つのダイヤルピアを設定して、同じユーザの複数のホストにサービスを提供できます。たとえば、同じダイヤルピアでtestuser@cisco.comとtestuser@webex.comを使用します。session target sip-uriを使用すると、着信Invite Req-URIのドメインのDNS解決がトリガーされ、セッションターゲットIPが動的に決定されます。

設定例：

ゲートウェイは、次のヘッダーを持つ2つのSIP Inviteを受信します。Invite sip:testuser@cisco.com:5060 SIP/2.0 Invite sip:testuser@webex.com:5060 SIP/2.0。incoming URIコマンドとユーザID定義がどちらもtestuserに一致するため、ダイヤルピア1でtestuser@cisco.comおよびtestuser@webex.comの着信SIP要求と一致します。コマンドvoice-class sip call-route urlが存在することは、この着信Inviteの要求URIに基づいて発信ダイヤルピアを評価することを意味します。ダイヤルピア2を照合するのは、ダイヤルピア1を照合した同じ理由 (testuserのユーザID) のためです。このダイヤルピアのセッションターゲットは、FQDNであるsession target sip-uriで定義された元のsip-uriです。DNS解決が行われた後、cisco.comおよびwebex.comをレイヤ3ルーティング用のIPに変更すると、ゲートウェイからメッセージが送信されます。

```
!  
ip host cisco.com 10.10.10.10  
ip host webex.com 10.10.10.10  
!  
voice class uri TEST-IN sip  
  user-id testuser  
!  
dial-peer voice 1 voip  
  description INCOMING dial-peer  
  incoming uri request TEST  
  session protocol sipv2  
  voice-class sip call-route url  
!  
dial-peer voice 2 voip  
  description OUTBOUND dial-peer  
  destination uri TEST  
  session protocol sipv2  
  session target sip-uri  
!
```

検証：

```
show voice class uri <uri-name>  
show voice class dial-peer provision-policy <number>
```

ダイヤルピアのワイルドカード

管理者は、番号文字列を含む着信および発信照合メカニズムを定義する際に、ダイヤルピアのワイルドカードを使用できます。これには destination-pattern、incoming called-number、e164-pattern-maps、answer-address、および prefix コマンドが含まれます。ダイヤルピアのワイルドカードは正規表現(regex)であり、ダイヤルピアの照合に関してより柔軟な設定が可能です。

ワイルドカードの表

文字	定義	例
*	これはダイヤルピアでキーパッド上の * (スター) のリテラル値です。	12345*
#	これはダイヤルピアでキーパッド上の # (シャープ) のリテラル値です。	8675309#
,	数字の間に1秒のポーズを挿入します。カンマは、角カッコ[]内で使用して、連続する範囲を分割することもできます。	9,,,55591[1-3,5-9]8675309
.	0 ~ 9、A ~ F、および *、#、+ のいずれかの値と照合するための正規表現文字 ダイヤルピアごとに最大15個のドット文字を定義できますが、管理者はCLIを使用して適切と思われる数だけ設定できます。 15個を超えるドットが必要な場合は、Tを使用してください。	2.... 91[2-9].[2-9].....
%	0回以上発生する前の数字の正規表現。	
+	文字列の先頭に使用された場合、リテラル + が E164 番号で使用されていることを意味します。 文字列内の他の場所で使用された場合、1回以上発生する前の数字の正規表現値です。	+19191112222
?	ゼロまたは1回発生する前の数字の正規表現。	(206)?5015111

		(0)?(1)?(1)?21933..
^	<p>角カッコの外側で使用された場合、文字列の開始を示す正規表現文字。</p> <p>角カッコの内側で使用された場合、除外または DO NO MATCH ステートメントとして扱われます。</p> <p>以降のバージョンでは、^なしの正規表現文字列を処理する際、ゲートウェイが自動的に^を想定するため、必要ありません。</p>	^8675309 91[^135]555
\$	文字列の終了を示す正規表現文字。	8675309\$
\	リテラル値を意味するエスケープ文字。	
[]	<p>角カッコは、1つの位置の文字の範囲を定義します。</p> <p>継続的な文字列を分割するためにカンマを使用する必要があります。</p>	[1-5]0000 [2,5-8]0000
()	丸カッコは、1つのセット内の文字のグループを定義します。	9(258)7777
T	<p>最大 32 桁の可変長一致。</p> <p>ルータはコールをルーティングする前に桁間タイムアウトが発生するまで待機します。</p> <p>桁間タイムアウトのデフォルト値は 10 秒で、音声ポートで timeouts inter-digit を使用して変更できます。</p> <p>管理者は、キーパッドで # を使用して桁間タイムアウトを終了できます。これは、コマンド dial-peer terminator をグローバルに設定して変更できます。</p> <p>T は T302 タイマーも参照します。</p>	9011T
-	角カッコ内で範囲を定義するために使用されます。	[5-9]1234

可能な正規表現の入力を表示するゲートウェイからの出力。

```
Gateway(config-dial-peer)# destination-pattern asdfqw4r3~2
Incorrect format for E.164 Number
regular expression must be of the form ^[][^\0-9,A-F#*.*+%()~]*T?(\$)?$
```

ダイヤルピアの状態

ダイヤルピアは、次の2つの動作状態のいずれかになります。

1. 2013 年以降
2. 停止

ダイヤルピアが有効な動作状態にあり、コールルーティングで使用するために適格であるためには、そのダイヤルピアがUP状態である必要があります。発信VoIPダイヤルピアの場合、これはコールをルーティングする有効な発信照合メカニズムとセッションターゲットが存在する可能性があることを意味します。発信POTSダイヤルピアの場合、有効な発信照合メカニズムと有効な音声ポートを設定できます。着信ダイヤルピアの場合のみ、有効な着信照合メカニズムを設定する必要があります。

ダイヤルピアにキープアライブメカニズムが設定され、リモートターゲットがそのキープアライブメカニズムのパラメータに失敗した場合、ビジョアウト状態が発生します。その後、ゲートウェイはダイヤルピアをビジョアウト状態に移行し、そのダイヤルピアはコールルーティングの決定に使用されなくなります。キープアライブメカニズムが再度実行されると、ゲートウェイはダイヤルピアを稼働状態に戻します。ダイヤルピアが発信ダイヤルピアとして選択され、このダイヤルピアがビジョアウト状態である場合、ゲートウェイは原因コード188でコールに失敗します。

動作状態とともに、次の管理状態があります。

1. 2013 年以降
2. 停止

管理者は、ダイヤルピアで shutdown コマンドを入力して、設定から削除せずにダイヤルピアを無効にすることができます。ダイヤルピアを再度有効にするには、no shutdown を入力します。

注：音声ポートがdown、shutdown、またはnot operationalの状態にあるダイヤルピアは、動作状態はupのままですが、Out StateはDownと表示されます。

検証

```
Gateway# show dial-peer voice summary
dial-peer hunt 0
```

TAG	TYPE	AD MIN	OPER	PREFIX	DEST-PATTERN	PRE FER	PASS THRU	SESS-TARGET	OUT STAT	PORT	KEEPALIVE
1	voip	up	up			0	syst				
777	voip	up	up		9...	0	syst	ipv4:10.50.244.2			
555	voip	up	down		555	0	syst				
888	pots	up	up		888	0			up	0/2/0	
999	pots	up	up		999	0			down	0/2/0	
123	voip	up	up		123	0	syst	ipv4:10.10.10.10			busyout

Virtual Routing and Forwarding (VRF) とダイヤルピア ハンテイング

VRFによる着信ダイヤルピアの照合

IOS 15.6(3)MおよびIOS-XE 16.3.1以降では、CiscoゲートウェイはVRF IDを使用して着信ダイヤルピアを照合できます。これを利用するには、管理者は着信ダイヤルピアをインターフェイスにバインドし、次にダイヤルピアを指定されたインターフェイスのVRF IDにバインドする必要があります。バインドが完了すると、着信コールはCiscoゲートウェイによってフィルタリングされ、パケットが受信されたインターフェイスのVRF IDと一致する、適格な着信ダイヤルピアだけが含まれます。ここから、着信ダイヤルピアは通常のダイヤルピア照合の処理順序に基づいて照合されます。

これらのIOS/IOS XEリリースより前のリリースでは、Ciscoゲートウェイは、フィルタリングを行わずに、通常の着信ダイヤルピア照合に基づいて着信を選択していました。これは、VRF1 コールを VRF2 ダイヤルピアによって照合できることを意味します。さらに、これらのリリースより前のH323およびSIPでは1つのVRFしかサポートされていなかったため、マルチVRF機能を使用しようとする、他の問題が発生します。音声アプリケーションに単一のVRFを使用する設定は、VRF対応設定と呼ばれていました。

VRF対応ドキュメント：[音声ゲートウェイ用のVRF対応H.323およびSIP](#)

完全なマルチVRFドキュメント：[Cisco Unified Border Elementコンフィギュレーションガイド - Cisco IOS XE 17.6以降](#)

VRF を使用した発信ダイヤルピアの照合

Cisco ゲートウェイには、ルート リークを設定することなく、VRF 全体でコールをブリッジする機能があります。これは、通常の発信ダイヤルピア一致の選択が満たされた場合、VRF1 の着信コールを VRF2 のダイヤルピアで発信にルーティングできることを意味します。ダイヤルピアグループを利用して、コールを同じ VRF 内に保持するように Cisco ゲートウェイに強制できます。

VRF およびダイヤルピア グループの設定例

この設定例には、2つの重複する IP 範囲と2つの重複する電話番号範囲を持つ VRF1 と VRF2 が示されています。

VRFバインディングを使用して正しい着信ダイヤルピアを照合し、Dial-peer Groupsを使用して正しいVRFバインド発信ダイヤルピアを照合します。8675309へのコールのSIPパケットが gig0/0/1.2に到達すると、ゲートウェイはVRF2 IDに基づいてすべての使用可能な着信ダイヤルピアをフィルタリングして除外します。これは、ダイヤルピア10を照合できないことを意味します。ここでディジットストリングを確認すると、ダイヤルピア20を照合できます。ダイヤルピア 20には、照合できる唯一の発信ダイヤルピアもダイヤルピア 20であることをゲートウェイに伝えるダイヤルピアグループがあります。このダイヤルピアグループを使用すると、ダイヤルピア 10の照合と、VRF1からVRF2へのコールの通過を回避できます。そこから、コールは通常どおり続行できます。

```

!
interface GigabitEthernet0/0/1.1
  description VRF1
  encapsulation dot1Q 10
  ip vrf forwarding VRF1
  ip address 10.10.10.10 255.255.255.0
!
interface GigabitEthernet0/0/1.2
  description VRF2
  encapsulation dot1Q 20
  ip vrf forwarding VRF2
  ip address 10.10.10.10 255.255.255.0
!
voice service voip
  no ip address trusted authenticate
  media-address voice-vrf VRF1
  media-address voice-vrf VRF2
  allow-connections sip to sip
  sip
!
voice class dpg 10
  description INBOUND VRF1 to OUTBOUND VRF1
  dial-peer 10 preference 1
!
voice class dpg 20
  description INBOUND VRF2 to OUTBOUND VRF2
  dial-peer 20 preference 1
!
dial-peer voice 10 voip
  description VRF1
  destination-pattern 8675309
  session protocol sipv2
  session target ipv4:10.10.10.20
  destination dpg 10
  incoming called-number 8675309
  voice-class sip bind control source-interface GigabitEthernet0/0/1.1
  voice-class sip bind media source-interface GigabitEthernet0/0/1.1
!
dial-peer voice 20 voip
  description VRF2
  destination-pattern 8675309
  session protocol sipv2
  session target ipv4:10.10.10.20
  destination dpg 20
  incoming called-number 8675309
  voice-class sip bind control source-interface GigabitEthernet0/0/1.2
  voice-class sip bind media source-interface GigabitEthernet0/0/1.2
!

```

検証

```

Gateway# show vrf brief | i VRF
  VRF1                1:1                ipv4                Gi0/0/1.1
  VRF2                2:2                ipv4                Gi0/0/1.2

```

```

Gateway# show dial-peer voice summary
TAG      TYPE  MIN  OPER PREFIX      DEST-PATTERN      FER THRU SESS-TARGET  STAT PORT  KEEPALIVE  VR
10       voip  up   up           8675309           8675309           0  syst  ipv4:10.10.10.20  STAT PORT  KEEPALIVE  VR
20       voip  up   up           8675309           8675309           0  syst  ipv4:10.10.10.20  STAT PORT  KEEPALIVE  VR

```

```

Gateway# show voice class dpg 10
Voice class dpg: 10      AdminStatus: Up
Description: INBOUND to OUTBOUND VRF1
Total dial-peer entries: 1

```

```

Peer Tag      Pref
-----
10            1
-----

```

```

Gateway# show voice class dpg 20
Voice class dpg: 20      AdminStatus: Up
Description: INBOUND to OUTBOUND VRF2
Total dial-peer entries: 1

```

```

Peer Tag      Pref
-----
20            1
-----

```

高度なコール ルーティング技術

長年にわたってビジネスニーズが増大するにつれて、会社が拡大し、より多くのDIDを必要とするようになり、企業管理者は基本的なダイヤルピアが適切にスケールに対応していないことを発見できます。対処が必要なオンオフ状況や、一般的なダイヤルピアが多すぎる可能性があります。数千のダイヤルピアを使用しても、管理やトラブルシューティングは容易ではありません。個別のCUCMサーバまたはコールエージェントごとにダイヤルピアを使用する場合、管理者は各デジタルストリングに対してダイヤルピアを設定する必要があるため、大量のダイヤルピアの問題が大きくなります。複数のSIPプロバイダーが1つのゲートウェイに接続している場合、または同じCUBEを使用する少数の異なるユーザが存在する場合、特定のテナントを分離するのは非常に困難になります。

シスコはこのフィードバックを受けて、これらの問題や他の問題に対処できる一連の項目を作成しました。ダイヤルピアグループ、音声クラステナント、宛先サーバグループ、e164-pattern-map、およびPOTSトランクグループを使用すると、管理者はリストされているすべての問題とここにはリストされていない問題を解決できます。

ダイヤルピア グループ

IOS 15.4(1)T と IOS-XE 3.11S でダイヤルピアグループが追加され、IOS 15.5(1)T と IOS-XE 3.14S で POTS ダイヤルピアがオプションとして追加されました。ダイヤルピアグループを使用

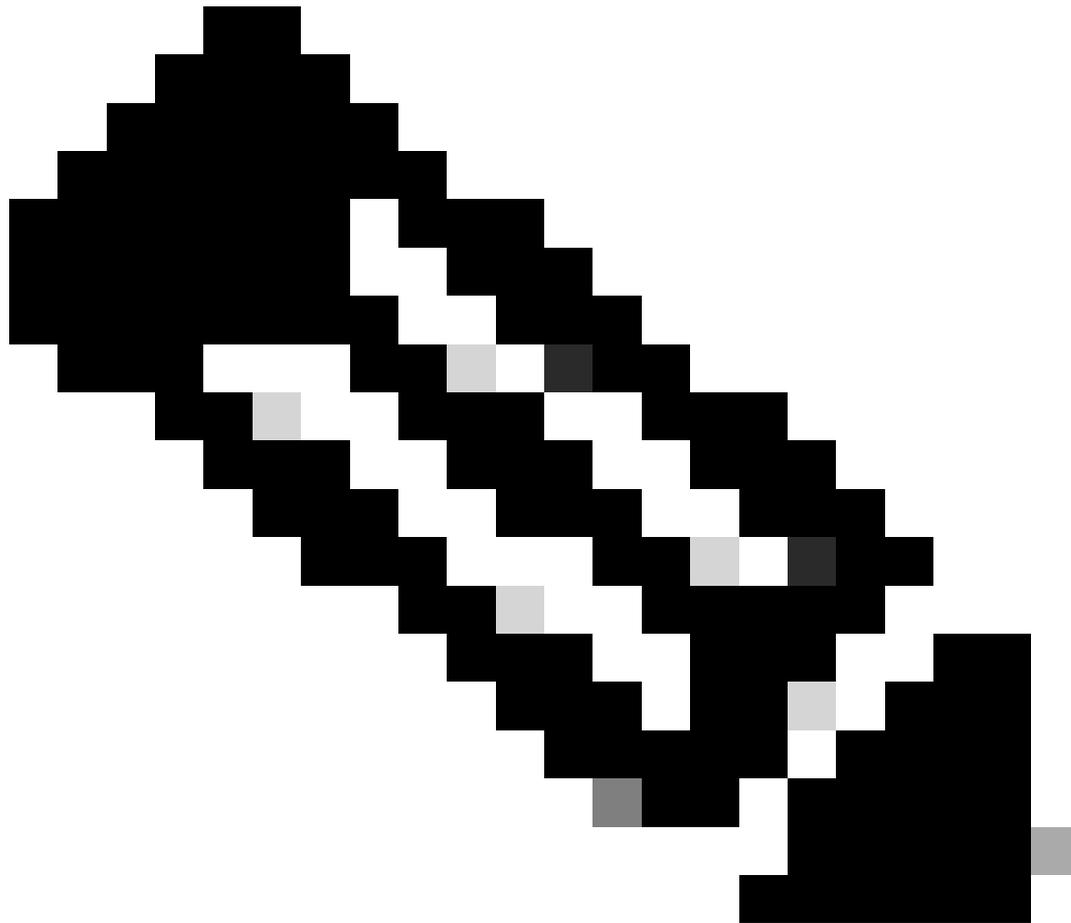
すると、管理者は一致した着信ダイヤルピアに基づいて発信ルーティングに正確なダイヤルピアを指定することができます。ダイヤルピアグループが設定された着信ダイヤルピアが一致すると、destination-patternが一致しなくても、コールはダイヤルピアグループで定義されたダイヤルピアを使用します。唯一の前提条件は、発信ダイヤルピアがUpであることです。そのため、発信照合メソッドを設定する必要がありますが、これはコールのルーティングに実際には使用されません。

ダイヤルピアグループを説明する最適な方法は、ルーティングテーブルで静的ルートのコンセプトに例えることです。これらは、着信から発信へのスタティックルーティングの決定であり、コールのルーティング方法をゲートウェイに正確に指示するため、ゲートウェイに対する推測の一部を取り除きます。

詳細なドキュメント：[Cisco Unified Border Elementコンフィギュレーションガイド – Cisco IOS XE 17.6以降](#)

設定例

この例では、着信者番号は8675309です。これは、incoming called-number ステートメントに基づいてダイヤルピア 1234 に一致します。このダイヤルピアは、ダイヤルピア2、3の順にコールをルーティングでき、ダイヤルピア2が失敗した場合は最後に1をルーティングできることを示すダイヤルピアグループで設定されます。ゲートウェイは、ダイヤルピアグループを介して明示的に指示されたとおりにダイヤルピア2をルーティングしようとします。



注：ダイヤルピア1、2、および3のdestination-patternは、8675309の着番号ではありません。これは適切であり、コールは引き続き問題なくルーティングされます。

「ダイヤルピアの状態」セクションで説明したように、発信照合ステートメントとして何か、またはすべて設定する必要があることに注意してください。この場合、宛先パターンはダイヤルピアをアップ動作状態にするためだけのものであり、そのコマンドのディジットストリングは評価されません。これは有効な宛先パターンであるため、destination-pattern AAAAのようなパターンを設定することをお勧めします。これは技術的には有効なダイヤルピアであるため、他のコールがこれに一致する可能性があります。したがって、AAAAにコールが着信する可能性は非常に低いため、AAAAディジットストリングは、ダイヤルピアグループを含む特定のシナリオ以外には使用できないことを意味します。

```
!  
dial-peer voice 1 voip  
description Server 1  
destination-pattern ^1234$  
session target ipv4:1.1.1.1
```

```

!
dial-peer voice 2 voip
  description Server 2
  destination-pattern ^5678$
  session target ipv4:2.2.2.2
!
dial-peer voice 3 voip
  description Server 3
  destination-pattern AAAA
  session target ipv4:3.3.3.3
!
voice class dpg 1
  description Dial-peer Group for specific called number 8675309
  dial-peer 2 preference 1
  dial-peer 3 preference 2
  dial-peer 1 preference 3
!
dial-peer voice 1234 voip
  description INCOMING dial-peer with DPG
  incoming called-number ^8675309$
  destination dpg 1
!

```

検証

```

Gateway# show voice class dpg 1
Voice class dpg: 1      AdminStatus: Up
Description: Dial-peer Group for specific called number 1234
Total dial-peer entries: 3

```

Peer Tag	Pref
2	1
3	2
1	3

E164-Pattern-Maps (発信ダイヤラマップ)

この機能により、管理者は多数の可能な番号の一致 (destination-patterns、incoming called-numberなど) を1つのパターンマップに結合することで、ダイヤルピアの総数を削減できます。発信ダイヤルピア e164-pattern-map のサポートは IOS 15.2(4)M と IOS-XE 3.7S で追加され、着信ダイヤルピア e164-pattern-map のサポートは IOS 15.4(1)T と IOS-XE 3.11S で追加されました。

e164-pattern-map は、CLI を使用して設定するか、事前設定して .cfg ファイルとして保存できます。その後 .cfg ファイルをゲートウェイのフラッシュに追加して、コマンドの残りの設定時に参照します。 .cfg ファイルでは 5000 エントリを利用できます。

どちらの設定方法のエントリも、すべての通常のダイヤルピアのワイルドカードを使用してさらに集約できます。

詳細なドキュメント：[Cisco Unified Border Elementコンフィギュレーションガイド - Cisco IOS XE 17.6以降](#)

CLI の設定例 - 発信者番号

```
!  
voice class e164-pattern-map 1  
  description E164 Pattern Map for calling numbers  
  e164 919574100.  
  e164 919574300.  
  e164 8675309  
!  
dial-peer voice 1 voip  
  description INBOUND Dial-peer based on CALLING #  
  incoming calling e164-pattern-map 1  
!  
dial-peer voice 11 voip  
  description OUTBOUND Dial-peer based on CALLING #  
  destination calling e164-pattern-map 1  
!
```

CLI の設定例 - 着信者番号

```
!  
voice class e164-pattern-map 2  
  description E164 Pattern Map for called 800 numbers  
  e164 91800T  
  e164 91855T  
  e164 91888T  
!  
dial-peer voice 2 voip  
  description INBOUND Dial-peer based on CALLED #  
  incoming called e164-pattern-map 2  
!  
dial-peer voice 22 voip  
  description OUTBOUND Dial-peer based on CALLED #  
  destination e164-pattern-map 2  
!
```

フラッシュの設定例

```
!  
voice class e164-pattern-map <tag>  
  description FILEPATH for E164 Pattern Map  
  url flash:<filepath>/e164-pattern-list.cfg  
!  
dial-peer voice ### voip  
  description E164 Pattern Map Dial-peer
```

```
incoming calling e164-pattern-map <tag>
!  
voice class e164-pattern-map load <tag>
```

検証

```
Gateway# show voice class e164-pattern-map 1
```

```
e164-pattern-map 1
```

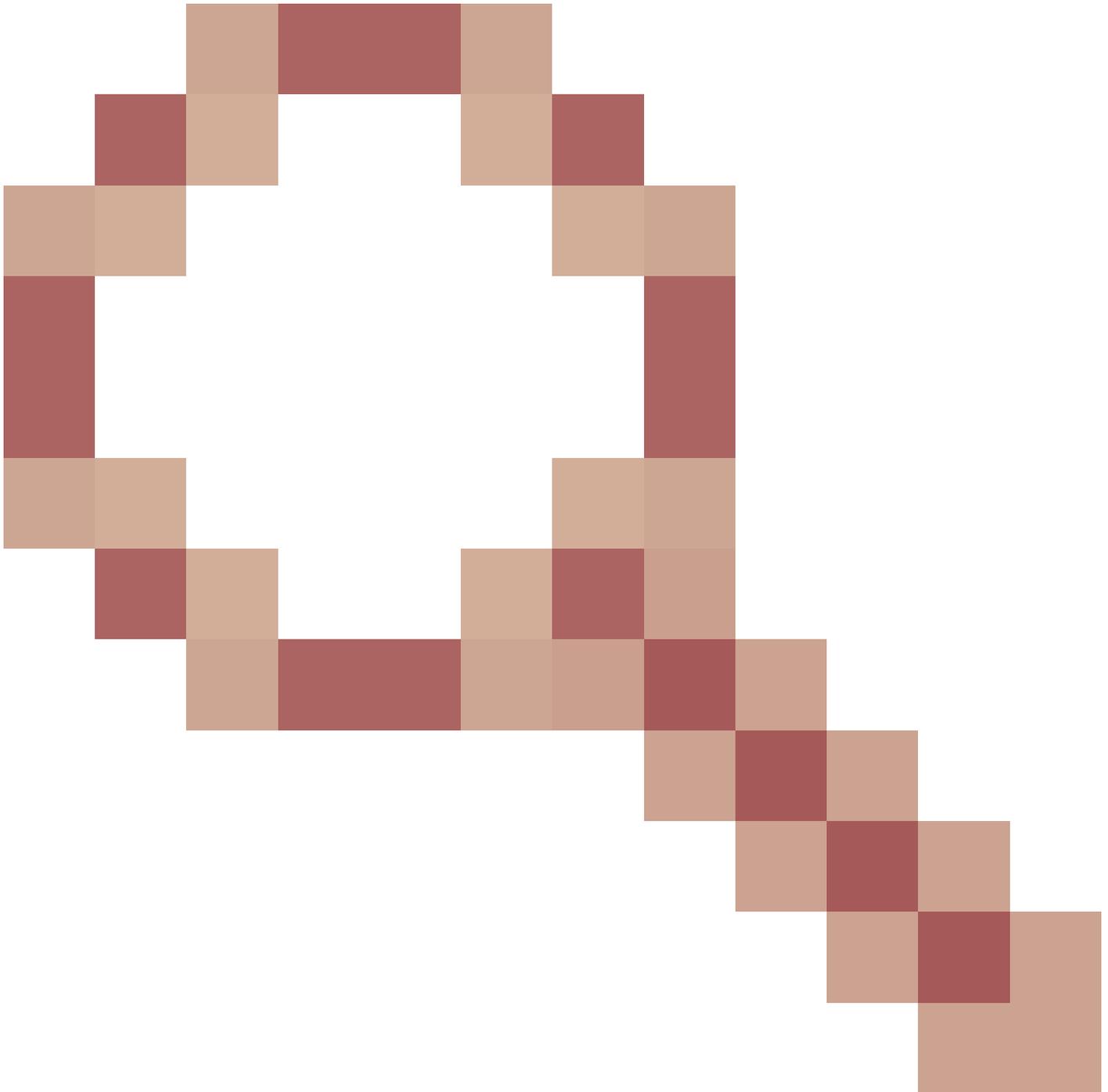
```
-----  
Description: CUCM phones  
It has 3 entries  
It is not populated from a file.  
Map is valid.
```

```
E164 pattern
```

```
-----  
8675309  
1...  
[2-5]...$
```

顕著な不具合

Cisco Bug ID [CSCva64393](#)



e164-pattern-mapでは、コンフィギュレーションファイルの最後の行が解析されません。

宛先サーバグループ

サーバグループにより、管理者は同じ VoIP ダイアルピアに複数の宛先（セッションターゲット）を設定できます。デフォルトでは、ソート順はサーバグループエントリで定義されている優先順位です。コマンド `hunt-scheme round-robin` を使用する場合、ラウンドロビンハンテイングを使用できます。サーバグループは、Cisco IOS 15.4(1)T および Cisco IOS XE 3.11S で追加されました。Cisco IOS XE 17.4.1 では、設定可能な `huntstop` エラーコードが `voice class-server` グループ設定に追加されました。つまり、404 Not Found などの単一のエラーコードを設定できます。SIP エラーは通常、デバイスがサーバグループ内の次のオプションを試行するようにトリガーします。サーバグループ内に `config huntstop 1 resp-code 404` を設定すると、ハンテイングを停止できます。また、`huntstop 1 resp-code 401 ~ 599` のような範囲で設定することもできます。

注：エントリの最大数は、サーバグループあたり5つです。

詳細なドキュメント：[Cisco Unified Border Elementコンフィギュレーションガイド – Cisco IOS XE 17.6以降](#)

設定例 – 標準

```
!  
voice class server-group 1  
  hunt-scheme round-robin  
  ipv4 10.50.244.2 port 5060 preference 1  
  ipv4 10.50.244.62  
  ipv6 2010:AB8:0:2::1 port 2323 preference 3  
  ipv6 2010:AB8:0:2::2 port 2222  
!  
dial-peer voice 1 voip  
  session protocol sipv2  
  destination-pattern 8675309  
  session server-group 1
```

!

検証

```
Gateway# show voice class server-group 1
Voice class server-group: 1
AdminStatus: Up OperStatus: Up
Hunt-Scheme: round-robin Last returned server:
Description:
Total server entries: 4
```

```
Pref Type IP Address IP Port
---- ---- -
1 ipv4 10.50.244.2 5060
0 ipv4 10.50.244.62
3 ipv6 2010:AB8:0:2::1 2323
0 ipv6 2010:AB8:0:2::2 2222
[..truncated..]
```

宛先サーバグループと OPTIONS キープアライブ

サーバグループは、通常のOut-of-dialog OPTIONSキープアライブメカニズムに従わないことに注意してください。サーバグループは、option-keepalive プロファイルと呼ばれる機能を利用します。この機能により、ゲートウェイは特定のサーバグループで定義されている各コールエージェントをモニタできます。

サーバグループでの option-keepalive の例

```
!
voice class server-group 1
  hunt-scheme round-robin
  ipv4 10.50.244.2
  ipv4 10.50.244.62
!
dial-peer voice 1 voip
  session protocol sipv2
  session server-group 1
  voice-class sip options-keepalive profile 1
!
```

検証

<#root>

Gateway#

```
show voice class sip-options-keepalive 1
```

```
Voice class sip-options-keepalive: 1          AdminStat: Up
Description:
Transport: system          Sip Profiles: 0
Interval(seconds) Up: 5    Down: 5
Retry: 5
```

Peer Tag	Server Group	OOD SessID	OOD Stat	IfIndex
-----	-----	-----	-----	-----
1	1		Active	87

```
Server Group: 1          OOD Stat: Active
OOD SessID  OOD Stat
-----
1           Active
2           Active
```

```
OOD SessID: 1          OOD Stat: Active
Target: ipv4:10.50.244.2
Transport: system      Sip Profiles: 0
```

```
OOD SessID: 2          OOD Stat: Active
Target: ipv4:10.50.244.62
Transport: system      Sip Profiles: 0
```

アウトバウンドプロキシ

SIPアウトバウンドプロキシ設定をvoice service voip、voice class tenant、またはダイヤルピア設定に追加して、レイヤ3 SIPパケットの宛先を指定できます。

つまり、ダイヤルピア上のセッションターゲットはSIPパケットの作成に使用できますが、発信プロキシはレイヤ3でパケットが送信される場所にすることができます。

```
!
voice service voip
  sip
    outbound-proxy dns:la01.sipconnect-us10.cisco-bcld.com
!
voice class tenant 100
  outbound-proxy dns:la01.sipconnect-us10.cisco-bcld.com
!
dial-peer voice 100 voip
  session target ipv4:192.168.1.1
  voice-class sip outbound-proxy dns:la01.sipconnect-us10.cisco-bcld.com
!
```

ダイヤルピアのデフォルト設定はvoice-class sip outbound-proxy systemであることに注意してください。この設定により、ダイヤルピアはグローバルなvoice service voip > sip設定を使用する可能性があります。

この動作をディセーブルにして、ダイヤルピアを強制的にフォールバックさせ、次の設定でダイ

ヤルピアごとにセッションターゲットをレイヤ3の宛先として使用できます。

```
dial-peer voice 777 voip
no voice-class sip outbound-proxy
```

POTS トランク グループ

トランク グループは、類似のシグナリング機能を持つ物理音声ポートのコレクションです。これらの機能を利用すると、設定する必要がある POTS ダイアルピアの総数を削減することができます。トランクグループは12.1(3)TでIOSに導入され、Cisco IOS XEのすべてのバージョンに存在します。

詳細なドキュメント：[ゲートウェイトランクおよびキャリアベースルーティングの拡張](#)

設定例

```
!
trunk group PSTN
  description PSTN voice-ports
!
trunk group FX0
  description FX0 voice-ports
!
voice-port 0/2/0
  trunk-group PSTN 1
!
voice-port 0/2/1
  trunk-group PSTN 2
!
voice-port 0/2/2
  trunk-group FX0 1
!
voice-port 0/2/3
  trunk-group FX0 2
!
dial-peer voice 1234 pots
  trunkgroup PSTN 1
  trunkgroup FX0 2
!
```

音声クラス テナント

Cisco IOS 15.6(2)TとCisco IOS XE 16.3.1では音声クラステナントが導入され、各テナントは独自の設定を個別に持つことができます。テナントは、テレフォニー プロバイダー、Cisco Unified Communications Manager (CUCM)、または管理者が特定のグローバル設定を行うその他のサードパーティ コール エージェントにすることができます。管理者は、まず音声クラス テナントを作成し、パラメータを定義します。次に、音声クラス テナントを特定のダイアルピアまたは選

扱に適用します。この新しい設定により、管理者はダイヤルピアとグローバル設定の他に、コールを別のレベルで制御できます。

17.8.1aでは、音声クラステナント設定をsip-listenコマンド（および適切なSIPコントロールバイインディングコマンド）で設定し、そのテナントの非セキュアポートまたはセキュアポートを定義できます。つまり、テナント1はUDP 5060 + VRF RedでセキュアでないSIPをリッスンし、テナント2はTCP TLS 5070 + VRF BlueでSIPをリッスンします。リスニングポート+バインド+オプションのvrf着信ダイヤルピアに基づいてテナントを照合した後、テナントが適用されているダイヤルピアに対してフィルタリングを行います。

詳細なドキュメント：[Cisco Unified Border Elementコンフィギュレーションガイド - Cisco IOS XE 17.6以降](#)

テナントのないコマンドプリファレンスの通常の順序

1. ダイヤルピア コマンド
2. グローバル コマンド (voice service voip および sip-ua)

テナントがあるコマンドプリファレンスの順序

1. ダイヤルピア コマンド
2. テナント コマンド
3. グローバル コマンド (voice service voip および sip-ua)

マルチテナントの設定例

777と999の2つのテナントがあります。これらの設定を少し異なる設定で行い、ダイヤルピアに適用しました。このことは、異なるダイヤルピアを使用するコールには、ダイヤルピアベースの設定と、テナント固有の設定があることを意味します。リストされているオプションは、音声クラステナントの機能の一部にすぎません。テナントで設定できる内容については、ドキュメントを参照してください。voice class uriなどの厳密な照合メカニズムを使用するか、特定の番号文字列を持つタギング番号を使用してテナントダイヤルピア照合を分離するか、VRFを設定して、テナントAがテナントBと重複したり、一致しないダイヤルピアと誤って一致したりしないようにすることをお勧めします。

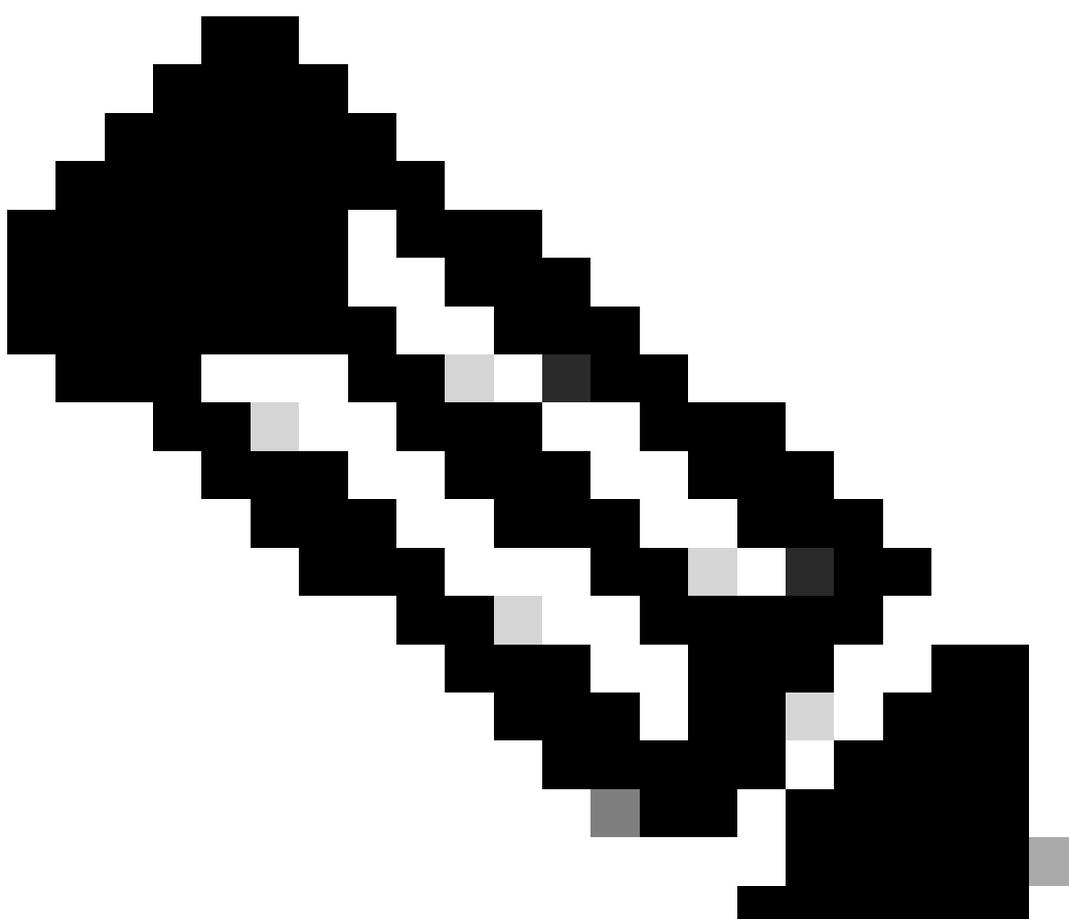
```
!  
voice class tenant 999  
  asymmetric payload full  
  bind control source-interface GigabitEthernet0/0/0.228  
  bind media source-interface GigabitEthernet0/0/0.228  
  g729 annex-a11  
!  
voice class tenant 777  
  sip-server ipv4:192.168.1.2  
  bind control source-interface Loopback0  
  bind media source-interface Loopback0  
  pass-thru content sdp  
!  
dial-peer voice 999 voip  
  destination-pattern 8675309  
  session protocol sipv2
```

```
incoming called-number 8675309
voice-class sip tenant 999
!
dial-peer voice 777 voip
destination-pattern 8675309
session protocol sipv2
session target sip-server
voice-class sip tenant 777
!
```

検証

現在、音声クラステナント設定を表示する個別のコマンドはありません。このコマンドは、実行コンフィギュレーションをテナント情報だけにフィルタリングするのに十分な場合があります。

```
show run | sec tenant
```



注: Cisco Bug ID [CSCvf28730](#)では、show sip-ua register statusが音声クラステナントでの SIP トランク登録のステータスを反映しません。

ILS URI コール CUBE (音声クラス ルート文字列)

ルート文字列は CUCM クラスター間ルックアップ サービス (ILS) で使用され、Cisco ゲートウェイが ILS サービスを実行する CUCM 9.5+ から受信した SIP INVITE に含まれるルート文字列によって VoIP コールをルーティングできるように設定できます。この機能は、Cisco IOS 15.3(3)M および Cisco IOS XE 3.10S で追加されました。ほとんどの ILS 接続は CUCM 間であり、管理者はクラスター間ランキング用の CUBE を含める必要はありません。ただし、CUBE を中央にして機能を実行する必要がある場合は、オプションがあります。CUBE に x-cisco-dest-route-string ヘッダーを送信するために、CUCM は SIP トランクに適用される SIP プロファイルで Send ILS Learned Destination Route String 設定を有効にする必要があります

詳細なドキュメント : [Enterprise Application Interoperability for H.323-to-SIP and SIP-to-SIP Configuration Guide, Cisco IOS Release 15M&T](#)

CUCM の設定例 – SIP - CUBE - SIP - CUCM

```
!  
voice service voip  
  sip  
  call-route dest-route-string  
!  
voice class route-string rt1  
  pattern london.uk.eu  
!  
voice class sip route-string rt2  
  pattern *.eu  
!  
voice class sip-hdr-passthru-list hdr1  
  passthru-hdr x-cisco-dest-route-string  
!  
dial-peer voice 1 voip  
  description INBOUND dial-peer  
  session protocol sipv2  
  voice-class sip pass-thru headers hdr1  
  incoming called-number .  
!  
dial-peer voice 2 voip  
  description OUTBOUND dial-peer  
  destination route-string rt2  
  session protocol sipv2  
  session target ipv4:172.16.104.178  
!
```

検証

```
show voice class route-string
```

従来のコール ルーティング技術

この項で説明する項目はレガシー技術と考えられています。これらを設定する機能は Cisco ゲートウェイ内にまだ存在しますが、今日の設定でこれらのコマンドを使用することはお勧めしません。このドキュメントでは、レガシー構成での作業中またはアップグレードの実行時に発生する可能性があるため、これらの問題のみを取り上げます。

DNIS マップ

DNIS-map は、現在の E164-pattern-map の前身と考えることができます。DNISマップは 12.2(2)XBでCisco IOSに追加され、Cisco IOS XEには常に存在していました。

DNISマップが設定されている場合は、より堅牢なe164-pattern-map機能に変換する価値がありません。

コマンド構文 : [Cisco IOS音声コマンドリファレンス-D ~ I](#)

設定例

```
!  
voice dnis-map 34  
  dnis 8675309  
!  
dial-peer voice 88 voip  
  dnis-map 34  
!
```

トランク グループ ラベル

トランクグループラベルはCisco IOS 12.2(11)Tで追加され、すべてのCisco IOS XEバージョンに存在します。トランクグループラベルの目的は、ダイヤルピアの照合を拡張するために使用できるという点で、キャリア ID に似ています。これは、POTS トランクグループ、VoIP および POTS ダイヤルピア、音声ソースグループ内の設定に利用できます。トランクグループラベルの使用は、現在のCiscoゲートウェイ設定ではほとんど見られません。

コマンド構文 : [Cisco IOS音声コマンドリファレンス-T ~ Z](#)

設定例

```
!  
dial-peer voice 112 pots
```

```
trunk-group-label source north3
trunk-group-label target east17
!
```

番号タイプ

ISDN Q.931 統合では、発信者番号または着信者番号と、Q.931 SETUP メッセージングの特定の ITU 番号タイプに基づいてダイヤルピアを照合する機能があります。これは、VoIP または POTS ダイヤルピアで numbering-type コマンドを使用して設定できます。numbering-type は単独で使用することはできず、destination-pattern、answer-address、または incoming called-number とともに使用する必要があります。つまり、ダイヤルピアの成功が着信および発信コールルーティングで考慮されるためには、着信/発信照合ステートメントの条件および番号タイプの両方が一致する必要があります。

番号一致は、照合メカニズムではなく、ダイヤルピア フィルタリング メカニズムと見なすことができます。これは、管理者プリファレンスが適用されていない場合、番号タイプ コマンドが適用されたダイヤルピアと適用されていないダイヤルピアは同じデフォルト プリファレンスのウェイトと見なされるためです。これは、他の照合メカニズムと一緒にダイヤルピアに適用された場合、両方の条件が true であればそのダイヤルピアにプリファレンスを追加する carrier-id とは異なります。

番号タイプの照合はCisco IOS 12.0(7)XR1で追加され、すべてのCisco IOS XEリリースに存在します。コラボレーション ネットワークに導入される従来の POTS ISDN 回線の減少に伴い、番号タイプの使用は現在の展開にはほとんど見られません。

コマンド構文 : [Cisco IOS音声コマンドリファレンス-K ~ R](#)

設定例

このダイヤルピアは、ISDN番号タイプがNationalの場合にだけ、4085150000 ~ 4085159999を照合できます。

```
!
dial-peer voice 408 voip
 numbering-type national
 destination-pattern 408515....
 session target ipv4:10.1.1.2
!
```

可能な番号タイプ :

Abbreviated	このネットワークでサポートされる完全な番号の省略表現
International	別の国のサブスクライバにアクセスするためにコールされる番号

National	同じ国でも、ローカル ネットワークの外部のサブスクライバにアクセスするためにコールされる番号
Network	サービス提供ネットワークに固有の管理またはサービス番号
Reserved	内線用に予約済み
Subscriber	同じローカル ネットワークのサブスクライバにアクセスするためにコールされる番号
[不明 (Unknown)]	番号のタイプはネットワークで不明

ダイヤルピア データ

データダイヤルピアはCisco IOS 12.2(13)Tで導入されました。このようなダイヤルピアの使用は、Ciscoゲートウェイでの着信データモデムコールのためでした。このダイヤルピアは着信方向専用で、現在の展開ではほとんど見られません。

コマンド構文：[Cisco IOS音声コマンドリファレンス-D ~ I](#)

設定例

```
!
dial-peer data 100 pots
  incoming called-number 100
!
```

音声ソースグループ

この機能は15.1(2)Tで追加されましたが、多くの最近の導入では実装されていません。IOS/CUBE用の他のセキュリティ方式は、通常、導入されます。

CUBEアプリケーションセキュリティの概要は、セクション4.2以降のこのホワイトペーパーで確認できます。

[Cisco Unified Border Element\(CUBE\)Management and Manageability Specification](#)

コマンド構文：[Voice Source-Group機能](#)

ダイヤルピアの権限

この設定により、管理者はダイヤルピアを制限して、着信接続のみ(term / terminate)または出力接続(orig / originate)を許可できます。これは、着信コールだけに使用する着信ダイヤルピアと、発信コールに使用する発信ダイヤルピアを明示的に設定するようなものです。すべてのダイヤルピアのデフォルトでは、着信接続と発信接続の両方が許可されます。このCLIは、最近の導入ではあまり使用されません。

```
Router(config)# dial-peer voice 1 voip
Router(config-dial-peer)# permission ?
  both allow both orig/term on this dialpeer
  none no orig/term allowed on this dialpeer
  orig allow only orig on this dialpeer
  term allow only term on this dialpeer
```

URI とディジット操作

コラボレーション導入のある時点で、管理者はディジットまたはURI/SIPヘッダーを操作する必要があります。Ciscoゲートウェイにはディジット操作の方法が数多くあり、管理者はディジットを操作する方法とタイミングを完全に制御できます。ただし、これは必ずしも簡単ではなく、さまざまなオプションに圧倒されたり、管理者がオプションの存在を知らない場合もあります。

POTS ダイヤルピアによるディジット操作

POTS ダイヤルピアには、VoIP ダイヤルピアにはない独自のディジット操作手法がいくつかあります。

最初は、destination-pattern の明確に定義された左揃えの数字の削除です。これは、POTS ダイヤルピアでコマンド no digit-strip を使用して無効にすることができます。

以下に例を挙げます。

この例では、9011Tがdestination-patternの文字列として定義されています。

これを行うと、90113227045555のコールを受信できます。これは発信コールルーティングのダイヤルピアと一致し、コールが音声ポートからルーティングされる前に、明示的に定義された9011桁が削除されます。

```
!
dial-peer voice 1 pots
  destination-pattern 9011T
  port 0/0/0:23
!
```

次の例は、番号削除が設定されていない設定を示しています。

同じ番号が呼び出されると、9011が送信されます (デフォルト) 。

```
!  
dial-peer voice 1 pots  
  destination-pattern 9011T  
  port 0/0/0:23  
  no digit-strip  
!
```

2つ目は、POTSダイヤルピアで転送するディジットの数を指定する機能です。

この例では、918005532447のコールをCUCMから受信します。この状況では、9を削除しますが、1で始まる残りの番号を送信します。

POTSダイヤルピアでforward-digitsコマンドを設定すると、送信する数字の数を正確に指定できます。

```
!  
dial-peer voice 1 pots  
  destination-pattern 918005532447  
  forward-digits 11  
  port 0/2/0  
!
```

最後に、POTSダイヤルピアでprefixコマンドを使用して、音声ポートからルーティングする前にコールに数字を追加できます。次の例では、コールを音声ポートから送信する前に、明示的に定義された91を削除し、番号の先頭に007を追加します。

```
!  
dial-peer voice 1 pots  
  destination-pattern 91T  
  prefix 007  
  port 0/1/0:15  
!
```

ボイス トランスレーション ルールとボイス トランスレーション プロファイルによるディジット操作

ボイス トランスレーション ルールは、数字の変換に使用される正規表現 (regex) です。トランスレーションルールとプロファイルは、12.0(7)XR1でCisco IOSに追加されました。トランスレーションルールはボイス トランスレーション プロファイルに適用され、ボイス トランスレーション

ンプロファイルはダイヤルピアまたは音声ポートに適用されます。トランスレーション ルールには、一致入力と変更出力が含まれます。番号に対する一致入力とともに、ISDN 計画およびタイプの一致と変更の入力があります。一致番号文字列、計画、およびタイプの組み合わせは、一致と見なされます。つまり、変換が行われるためには、定義されているすべての一致入力 が true である必要があります。

トランスレーション ルールには、ISDN、SIP、H323 シグナリング プロトコルの着信者番号、発信者番号、リダイレクト着信、リダイレクト ターゲット、およびコールバック番号を変更する機能があります。トランスレーションルールはトップダウン検索に基づいて照合するため、ルールの順序が最も重要です。上位のルールで一致が見つかった場合、ゲートウェイは即座に検索を停止し、変換を処理します。トランスレーションルールは、testuser@10.10.10などの数字以外のsipヘッダーは変更できません。この操作では、SIPプロファイルを使用します。

トランスレーション ルールを使用して Cisco ゲートウェイでコールをブロックできます。

トランスレーション プロファイルの選択プリファレンス

1. 音声ポートの着信ボイス トランスレーション プロファイル
2. トランク グループがシリアル インターフェイスに適用された着信ボイス トランスレーション プロファイル
3. 着信ダイヤルピアの着信ボイス トランスレーション プロファイル
4. 音声サービス POTS によって定義された着信ボイス トランスレーション プロファイル
5. グローバル「voip-incoming translation-profile」によって定義された着信ボイス トランスレーション プロファイル
6. 音声サービス POTS によって定義された発信ボイス トランスレーション プロファイル
7. 発信ダイヤルピアの発信ボイス トランスレーション プロファイルまたは変換発信
8. シリアル インターフェイスに適用されたトランク グループの発信ボイス トランスレーション プロファイル
9. 音声ポートの発信ボイス トランスレーション プロファイル

ダイヤルピアの正規表現とワイルドカードに加えて、トランスレーション ルールには独自の正規表現文字があります。

文字	定義
*	トランスレーション ルールで使用された場合、これは前の文字の 0 以上の正規表現です。 リテラル*に一致させるには、エスケープ文字*を使用します。
\	トランスレーション ルールでセットをエスケープするために一般的に使用されます (\\)。
&	アンパサンドは、変更セットの最初の一致セットで一致したものを取り込むために使用されます。

()	丸カッコで囲まれた項目は、1つのセットと見なされます。
^	<p>文字列の明示的な開始を定義します。</p> <p>ダイヤルピアのトランスレーションルールとは異なり、文字列の開始を定義しません。</p> <p>これは、^を使用せずに文字列を定義すると、入力文字列の任意の場所と一致する可能性があり、番号の途中で不要な変換が発生する可能性があることを意味します。</p>

変更セット

- セットは、\0、\1、\2のように指定します。
- \0 は、最初の一致セットの間にあるいずれかのものと一致します。これは、アンパサンド文字&を使用して実行することもできます。
- \1 は、一致セット内の最初の () のセットと一致します。
- \2 は、一致セット内の 2 番目の () のセットと一致します。
- 以下同様に続きます。

2つのセットを含むトランスレーションルールの例

この例では、番号000111000222を調べることができます。

この数値から0を削除して、最終的な111222の数値を得る場合。

これを行うには、セット1と2を設定して、それぞれ111と222を取得し、0をドロップします。

```
!
voice translation-rule 333
 rule 1 /000\((111\)000\((222\)\/ /\1\2/
!
voice translation-profile SET-EXAMPLE
 translate called 333
!

Gateway# test voice translation-rule 333 000111000222
Matched with rule 1
Original number: 000111000222   Translated number: 111222
Original number type: none       Translated number type: none
Original number plan: none       Translated number plan: none
```

着信者番号のダイヤルパターンから9を削除する例

```
!
voice translation-rule 9
 rule 1 /^9\(.*\)/ /\1/
!
voice translation-profile STRIP-9
 translate called 9
```

```
!  
dial-peer voice 9 voip  
  translation-profile outgoing STRIP-9  
!  
voice-port 0/0/0  
  translation-profile outgoing STRIP-9  
!  
  
Gateway# test voice translation-rule 9 918675309  
Matched with rule 1  
Original number: 918675309      Translated number: 18675309  
Original number type: none      Translated number type: none  
Original number plan: none      Translated number plan: none
```

着信者番号を4桁に切り捨てる

```
!  
voice translation-rule 4  
  rule 1 /.*\(...\) / \1/  
!  
voice translation-profile STRIP-T0-4  
  translate called 4  
!  
  
Gateway# test voice translation-rule 4 8675309  
Matched with rule 1  
Original number: 8675309      Translated number: 5309  
Original number type: none      Translated number type: none  
Original number plan: none      Translated number plan: none
```

着信者番号からプラス + を削除する

```
!  
voice translation-rule 999  
  rule 1 /\+\.*/ \1/  
!  
voice translation-profile STRIP-PLUS  
  translate called 999  
!  
  
Gateway# test voice translation-rule 999 +8675309  
Matched with rule 1  
Original number: +8675309      Translated number: 8675309  
Original number type: none      Translated number type: none  
Original number plan: none      Translated number plan: none
```

トランスレーション ルールは、最初にトランスレーション プロファイルに適用せずにダイヤルピアに直接適用することもできます。

```
!  
voice translation-rule 1  
  rule 1 /1234/ /8678309/  
!  
voice translation-rule 2  
  rule 2 /^4...$/ /1408515\0/  
!  
dial-peer voice 1 voip  
  translate-outgoing called 1  
!  
dial-peer voice 2 voip  
  translate-outgoing calling 2  
!
```

トランク グループのトランスレーション プロファイル

```
!  
trunk group <name>  
  translation-profile incoming <profile-name>  
  translation-profile outgoing <profile-name>  
!
```

ボイス トランスレーション ルールとボイス トランスレーション プロファイルのデバッグ

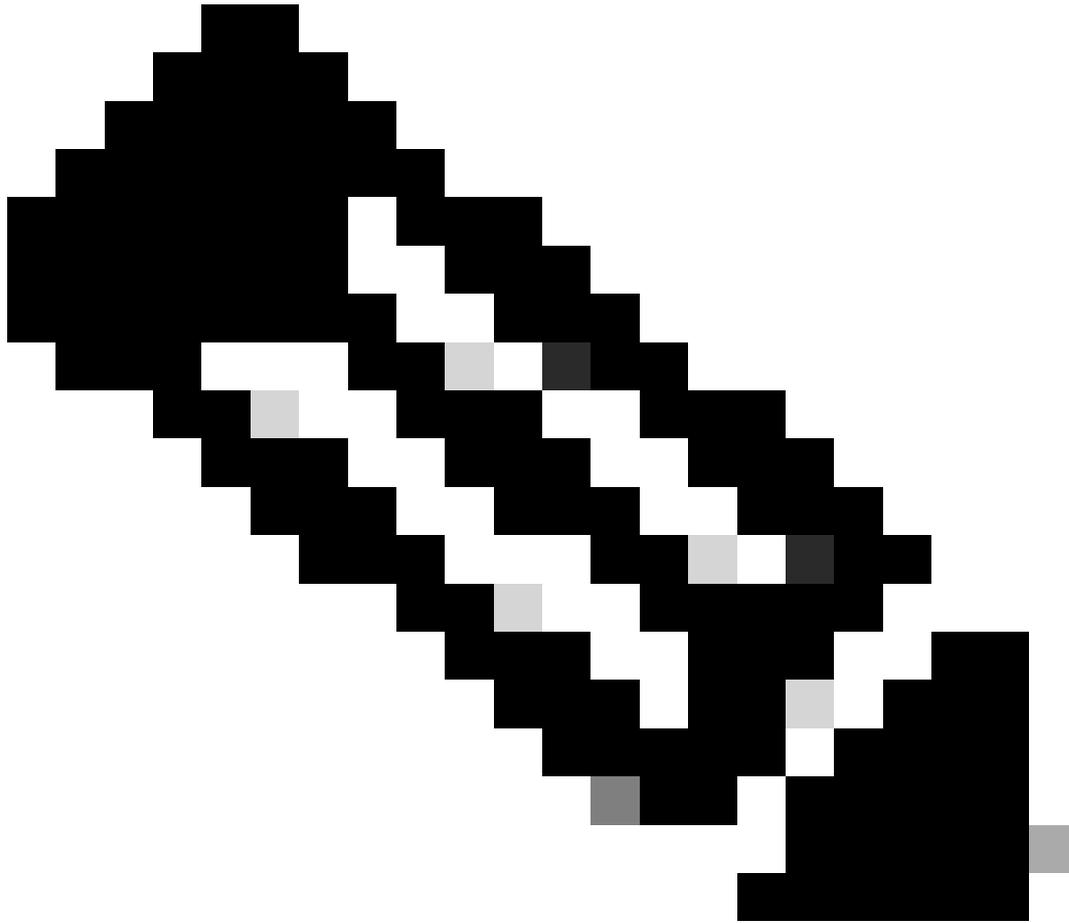
```
debug voip ccapi inout  
debug voice translation  
debug dialpeer  
test voice translation-rule <number> <string> type <type> plan <plan>
```

音声クラス e164 変換

voice class e164-translation機能は、より新しいCisco IOS XEの機能です。この機能を使用すると、管理者は、一致ステートメントのリストを作成し、フラッシュまたはネットワークディレクトリからコンフィギュレーションファイルを介してロードされるステートメントを変更できます。これは、このドキュメントで説明したe164-pattern-map機能の概念に似ています。これにより、管理者はコンフィギュレーションファイル内で最大100の変換を設定し、それらを単一の変換プロファイルに適用できます。詳細については、『[Cisco IOS 音声コマンドリファレンス](#)』を参照してください。

.cfgファイルの構文は次のとおりです。

```
pattern1_to_be_matched<tab>replaced_pattern<space><enter>  
pattern1_to_be_matched<tab>replaced_pattern<space><enter>
```



注：末尾のスペースは非常に重要で、追加のフォーマット手順を実行しないとインポートが失敗する可能性があります。

サンプル.cfg

```
+111111 8897
+222222 8312
928747 +123456789
737362 +987654321
```

このファイルでは、次のように参照されます。

```
voice class e164-translation 164
url ftp://username:password@10.10.10.10/sample.cfg
```

ここで、トランスレーションプロファイルに通常どおり適用してから、通常のトランスレーションプロファイル構文を使用してダイヤルピアに適用します。

```
voice translation-profile e164
  translate calling voice-class e164-translation 164
  translate called voice-class e164-translation 164
```

show voice class e164-translation e164-translation-numberコマンドを使用すると、トランスレーションプロファイルの内容を表示できます。

ISDN マップによるディジット操作

ISDN マップは、数字を変更するための古い技術です。これはCisco IOS 12.0(6)Tで追加されたもので、ボイストランスレーションルールやボイストランスレーションプロファイルほど堅牢ではないため、ほとんどの新しい設定ではこの機能が利用されていません。ISDN マップは、シリアルインターフェイスで定義されます。

設定例

```
Serial0/0/0:23
 isdn map address ^911 plan isdn type unknown
 isdn map address ^1..... plan isdn type national
 isdn map address ^2..... plan isdn type national
 isdn map address ^3..... plan isdn type national
 isdn map address ^4..... plan isdn type national
 isdn map address ^5..... plan isdn type national
 isdn map address ^6..... plan isdn type national
 isdn map address ^7..... plan isdn type national
 isdn map address ^8..... plan isdn type national
 isdn map address ^9..... plan isdn type national
```

番号拡張 (num-exp) によるディジット操作

ISDNマップと同様に、番号拡張はCisco IOS 11.3(1)Tで追加された古い技術で、新しいネットワークではあまり使用されません。この機能は、ボイストランスレーションルールとボイストランスレーションプロファイルが作成される前に追加されました。番号拡張は、Cisco ゲートウェイのすべてのダイヤルピアに適用される数字のグローバルな変更です。ダイヤルピアが照合された後、コールが次のコールエージェントに送信される直前に、変更が着信者番号に適用されます。

設定例

```
num-exp 4... 18005554...
```

着信/発信 SIP プロファイル

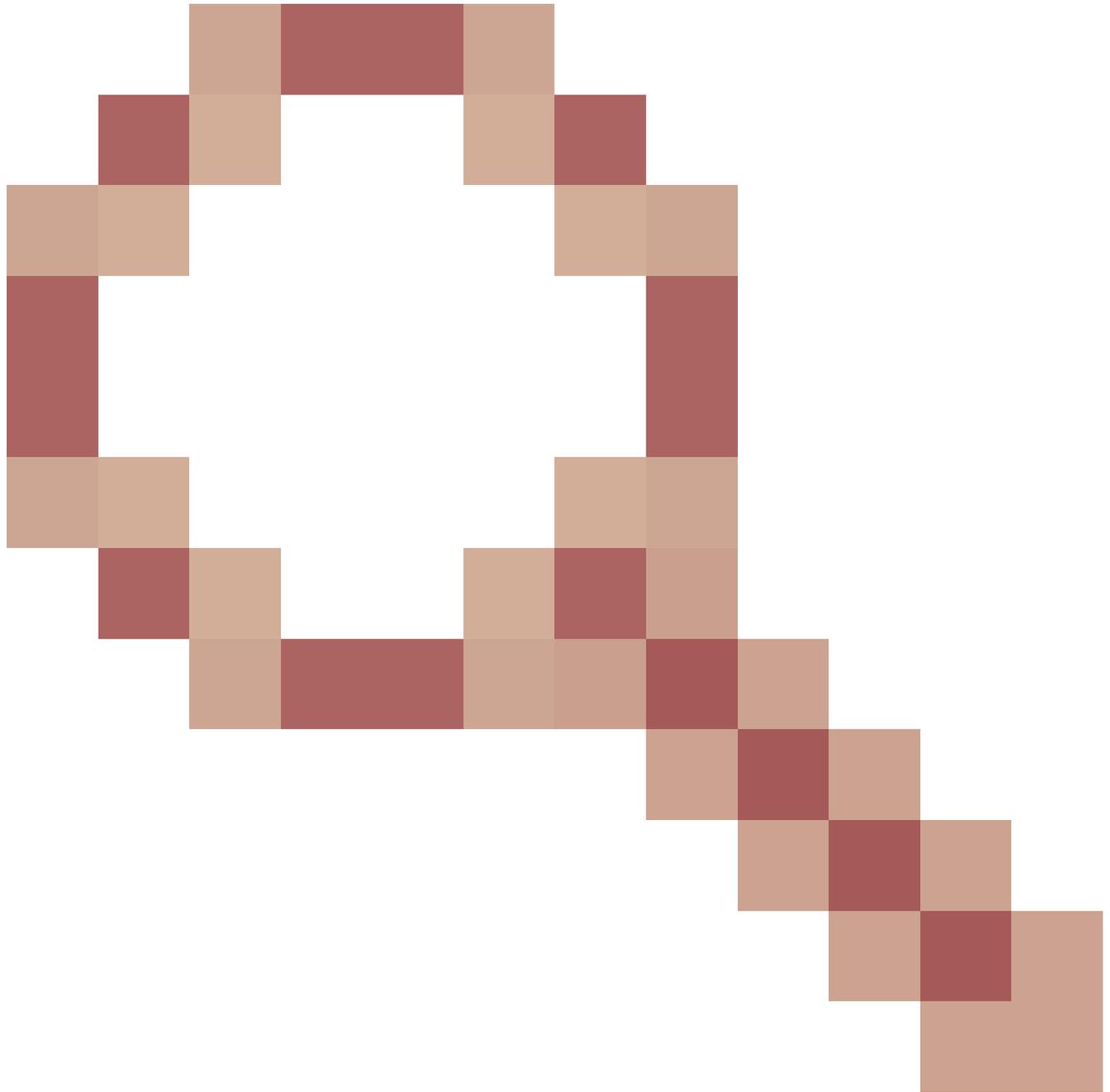
SIPプロファイルは堅牢な正規表現(regex)照合ステートメントであり、管理者はSIPプロファイルを使用して、SDPヘッダーとSIPヘッダーを含むSIPメッセージのあらゆる側面を変更できます。これらは、グローバル、ダイヤルピアごと、またはテナントごとに有効にできます。SIPプロファイルは、Cisco IOS 15.4(2)TおよびCisco IOS XE 3.12S以降の着信の変更に使用できます。SIPプロファイルは非常に堅牢であるため、このドキュメントではいくつかの具体的な例のみを取り上げます。SIPプロファイルには、カスタムSIPヘッダーをIOS 15.5(2)TおよびIOS-XE 3.13Sで変更または追加する機能も追加されています。

着信 SIP プロファイルと発信 SIP プロファイルに関する重要なポイント

- 着信 SIP プロファイルは、CUBE がコール ルーティングのメッセージを処理する前にメッセージを変更します。
- 発信 SIP プロファイルは、CUBE がコール ルーティングを処理した後、メッセージがネクスト ホップに送信される前にメッセージを変更します。

sip-profile の設定に関するその他の注意：

- SIP プロファイルは m=image SDP 属性を操作できません。この機能拡張は、Cisco Bug ID [CSCsr20474](#)に記載されています。また、SIPプロファイルではSDPに対する値の削除や追加を行うことがで



きません。これらの値を変更できるのはユーザだけです。ただし、値全体を指定してから、出力をスペースなしの空の引用符のセットに設定することで、SDP値をNULL値に変更できます。

- 音声クラス SIP プロファイル内でコマンドを入力する場合、入力された現在のコマンドがすでに存在するかどうか、またはそのコマンドのバージョンがすでに存在するかどうかを確認するためのチェックは実行されません。管理者がSIPプロファイルに行を7回貼り付けた場合、その行は実行コンフィギュレーションに7回表示されます。SIP プロファイルを編集する場合は、複数のコマンドが存在することを回避するために、変更したコマンドを削除してから、新しいコマンドを入力することを推奨します。

詳細なドキュメント：[Cisco Unified Border Elementコンフィギュレーションガイド – Cisco IOS XE 17.6以降](#)

SIPプロファイルテストツール：[SIPプロファイルテストツール](#)

着信/発信 SIP プロファイルのサンプル構文

```

!
voice class sip-profiles <number>
  request <message-type> sip-header <header> modify "match-pattern" "replace-pattern"
  request <message-type> sip-header <header> add "add-pattern"
  request <message-type> sip-header <header> remove

  request <message-type> sdp-header <header> modify "match-pattern" "replace-pattern"
  request <message-type> sdp-header <header> add "add-pattern"
  request <message-type> sdp-header <header> remove

  response <number> sip-header <header> modify "match-pattern" "replace-pattern"
  response <number> sip-header <header> add "add-pattern"
  response <number> sip-header <header> remove

  response <number> sdp-header <header> modify "match-pattern" "replace-pattern"
  response <number> sdp-header <header> add "add-pattern"
  response <number> sdp-header <header> remove
!

```

番号による着信/発信SIPプロファイルの例

```

voice class sip-profiles 200
  rule 1 response ANY sip-header Remote-Party-ID modify "match-pattern" "replace-pattern"
  rule 2 response ANY sdp-header Audio-Attribute modify "match-pattern" "replace-pattern"

```

発信 SIP プロファイル アプリケーション メソッド

<#root>

! Global Application

```

voice service voip
  sip
    sip-profiles <number>
!

```

! Tenant Application

```

voice class tenant <tag>
  sip-profiles <tag>
!

```

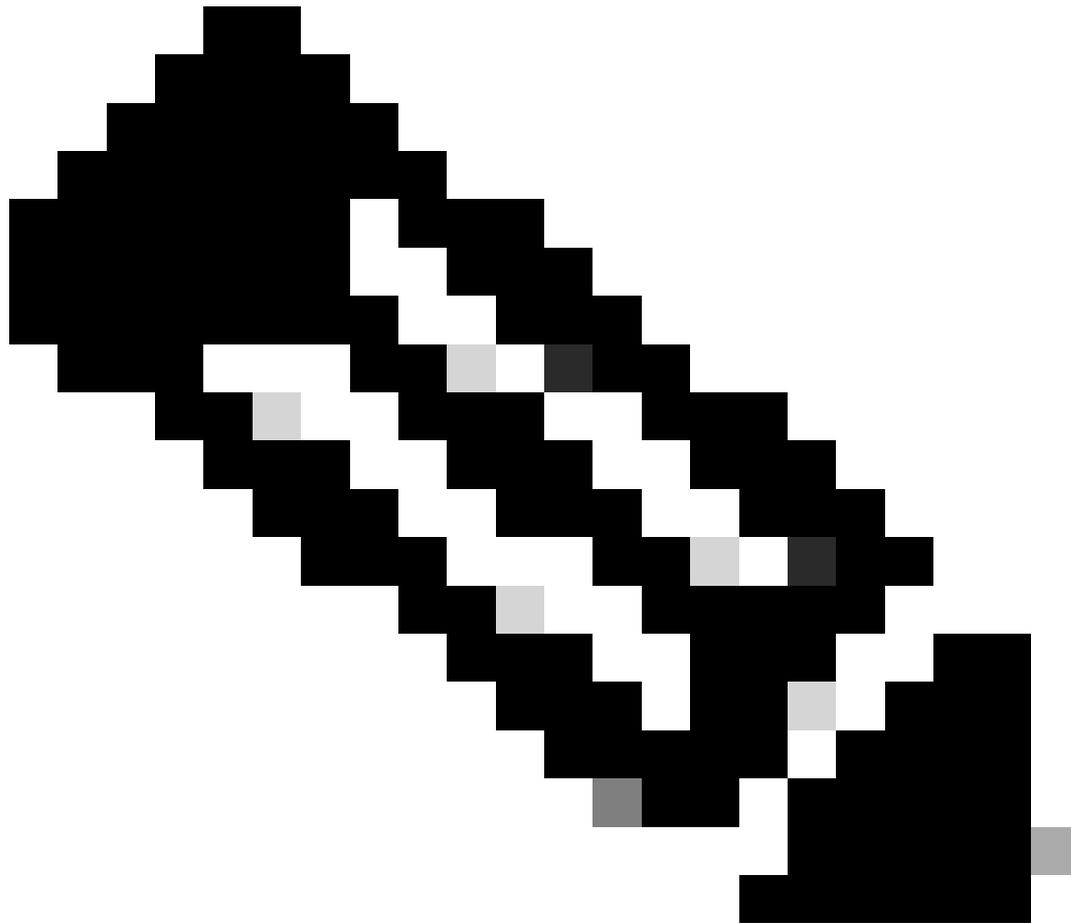
! Dial-peer Application

```

dial-peer voice <tag> voip
  voice-class sip profiles <number>
!

```

着信 SIP プロファイル アプリケーション メソッド



注：グローバルアプリケーション、テナント、またはダイヤルピアアプリケーションを使用するかどうかにかかわらず、voice service voip sipでsip-profile inboundを有効にする必要があります。

```
<#root>
```

```
! Global Application
```

```
voice service voip
  sip
    sip-profiles inbound
    sip-profiles <number> inbound
!
```

```
<#root>
```

```
! Tenant Application
```

```
voice service voip
  sip
    sip-profiles inbound
!
voice class tenant <tag>
  sip-profiles <tag> inbound
!
```

<#root>

! Dial-Peer Application

```
voice service voip
  sip
    sip-profiles inbound
!
dial-peer voice <tag> voip
  voice-class sip profiles <number> inbound
!
```

OPTIONSキープアライブメッセージを変更するSIPプロファイルの例。

```
!
voice class sip-options-keepalive 200
  transport tcp tls
  sip-profiles 299
!
```

URIのホスト、ドメイン、または両方の部分を変更するSIPプロファイルの例。

<#root>

! Host

```
!
voice class sip-profiles 1
  request ANY sip-header Contact modify "sip:(.*)@" "sip:8675309@"
!
```

! Domain

```
!
voice class sip-profiles 2
  request ANY sip-header Contact modify "10.67.138.241:5060" "cisco.com"
!
```

! Note: Port is optional

!

! Modify Both User and Host

```
!  
voice class sip-profiles 3  
  request ANY sip-header Contact modify "sip:(.*)>" "sip:8675309@cisco.com>"  
!
```

Diversionヘッダーを追加、変更、または削除するSIPプロファイルの例。

```
<#root>  
  
! Add  
  
!  
voice class sip-profiles 777  
  request INVITE sip-header Diversion add "Diversion: <sip:1234@cisco.com>"  
!  
!  
  
! Modify  
  
!  
voice class sip-profiles 888  
  request INVITE sip-header Diversion modify "sip:(.*)>" "sip:1234@cisco.com>"  
!  
!  
  
! Remove  
  
!  
voice class sip-profiles 999  
  request INVITE sip-header Diversion remove  
!
```

SIPヘッダーの発信者ID名部分を変更するSIPプロファイルの例。

```
!  
voice class sip-profiles 123  
  request INVITE sip-header From modify "\".*\\"" "\"TEST CLID*\\""  
!
```

183 セッション進行中を 180 呼び出し中に変更する SIP プロファイルの例。

```
!  
voice class sip-profiles 789  
  response 183 sip-header SIP-StatusLine modify "SIP/2.0 183 Session in Progress" "SIP/2.0 180 Ringing"  
!
```

プロバイダーとの一方向または方向なしのオーディオ相互運用性のSIPプロファイルの例。

```
<#root>
```

```
!  
voice class sip-profiles 200  
  request ANY sdp-header Audio-Attribute modify "a=inactive" "a=sendrecv"  
  request ANY sdp-header Audio-Connection-Info modify "0.0.0.0" "10.10.10.10"  
!  
! where 10.10.10.10 is CUBE's provider facing IP address
```

相互運用性の問題のために UPDATE メソッドを削除する SIP プロファイルの例。

```
!  
voice class sip-profiles 200  
  request ANY sip-header Allow-Header modify ", UPDATE" ""  
!
```

SIP プロファイル内でのセットの使用を示す SIP プロファイルの例。これは、ボイス トランスレーション ルールの項で説明したセットの概念と同じです。

```
!  
voice class sip-profiles 1  
  request ANY sip-header Contact modify "sip:(.*)@" "sip:\1@"  
!
```

SIP プロファイルでの IF ロジックと改行の設定。

SIP プロファイルでは改行がサポートされていますが、非常に特殊な使用例が1つだけあります。SIP プロファイルには If、Then、Else ロジックがないため、別のヘッダーからの入力に基づいて1つのヘッダーに対して変更を実行する方法があります。たとえば、管理者は、FROM ヘッダーに 1234@cisco.com が含まれている場合にのみ Diversion ヘッダーを変更する必要があります。改行を使用して、SIP プロファイル内で IF ステートメントをスプーフィングできます。次の設定例を参照してください。From ヘッダー内の任意のドメインで 1234 と一致しています。次に、最初のセットを取り込み、新しい改行 \x0D\x0AD を追加します。最後に、必要なヘッダーを追加します。この方法では、ヘッダーの追加のみが可能です。別のヘッダーを変更することはできません。そのため、この方法では、管理者が以前に達成しようとした要件の一部しか満たすことができません。

```
!  
voice class sip-profiles 1  
  request INVITE sip-header From modify "(.*sip:1234@.*)" "\1\x0D\x0ADiversion: <sip:5678@example.com>"  
!
```

ORロジックを使用したSIPプロファイルの例。

```
!  
voice class sip-profiles 123  
  request ANY sdp-header Audio-Attribute modify "(a=sendonly|a=recvonly|a=inactive)" "a=sendrecv"  
  response ANY sdp-header Audio-Attribute modify "(a=sendonly|a=recvonly|a=inactive)" "a=sendrecv"  
!
```

SIPプロファイルによるレイヤ7 SIPインスペクションの例

<#root>

Usage

10.21.15.10 replace with private IP of CUBE
a.b.c.d replace with public IP

Inbound from ITSP Layer 7 Fixup

```
!  
voice class sip-profiles 888  
  request INVITE sip-header SIP-Req-URI modify "@.*;" "@10.21.15.100;"  
!  
voice service voip  
  sip  
  sip-profiles inbound  
!
```

Outbound Layer 7 Fixup

```
!  
voice class sip-profiles 777  
  request ANY sip-header Contact modify "<sip:(.*)@10.21.15.100:5060>" "<sip:\1 a.b.c.d:5060>"  
  response ANY sip-header Contact modify "<sip:(.*)@10.21.15.100:5060>" "<sip:\1 a.b.c.d:5060>"  
  request ANY sip-header Via modify "SIP(.*) 10.21.15.100(.*)" "SIP\1 a.b.c.d\2"  
  request ANY sdp-header Session-Owner modify "(.*IP4 ).*" "\1a.b.c.d"  
  request ANY sdp-header Connection-Info modify "IN IP4 10.21.15.100" "IN IP4 a.b.c.d"  
  request ANY sdp-header Audio-Connection-Info modify "IN IP4 10.21.15.100" "IN IP4 a.b.c.d"  
  response ANY sdp-header Session-Owner modify "(.*IP4 ).*" "\1a.b.c.d"  
  response ANY sdp-header Audio-Connection-Info modify "IN IP4 10.21.15.100" "IN IP4 a.b.c.d"  
  response ANY sdp-header Connection-Info modify "IN IP4 10.21.15.100" "IN IP4 a.b.c.d"  
  request ANY sip-header Remote-Party-ID modify "<sip:(.*)@10.21.15.100>" "<sip:\1 a.b.c.d>"  
  response ANY sip-header Remote-Party-ID modify "<sip:(.*)@10.21.15.100>" "<sip:\1 a.b.c.d>"  
!
```

Apply to dial-peers for the side of the CUBE facing the ITSP

```
!  
dial-peer voice 1 voip  
  voice-class sip profiles 777
```

```
voice-class sip profile 888 inbound
!
dial-peer voice 2 voip
voice-class sip profiles 777
voice-class sip profile 888 inbound
!
```

SIP Copylist

SIP CopylistsはSIPプロファイルの拡張で、ゲートウェイはコールの内部レッグからヘッダーをコピーして、外部レッグの出力SIPメッセージの別の場所に貼り付けることができます。SIP Copylistのサポートは、Cisco IOS 15.1(3)TおよびCisco IOS XE 3.6Sで追加されました。これは、コールの内部レッグからの他のヘッダーに基づいて動的ヘッダーを作成する非常に強力な方法です。

最も一般的な使用例では、diversion や p-asserted-id のような異なるヘッダーに FROM ヘッダーを動的にコピーし、ユーザ部分の値が from のユーザになるようにします。これはほとんどの場合、認証と発信者 ID の目的で行われます。

詳細なドキュメント：[Cisco Unified Border Elementコンフィギュレーションガイド – Cisco IOS XE 17.6以降](#)

SIP Copylist の例

```
<#root>
```

```
!
! Create Copylist to copy the FROM header on the inbound message specified later.
!
voice class sip-copylist <number>
sip-header From
!
! Apply this to the inbound dial-peer of the call.
!
dial-peer voice <tag> voip
voice-class sip copy-list <number>
!
! Create SIP Profile to take From (peer-header) stored as variable "u01" and apply to a header of choice
! This example modifies the user portion of the Contact by copying the user portion of the From header t
!
voice class sip-profiles <number>
request INVITE peer-header sip From copy "<sip:(.*)@" u01
request INVITE sip-header Contact modify "<sip:(.*)>" "<sip:\u01@10.50.244.2>"
!
! Apply the SIP profile to an outbound dial-peer
!
dial-peer voice <tag> voip
voice-class sip profiles <number>
```

!

SIP プロファイルと Copylist のデバッグ

```
debug voip ccapi inout
debug ccsip mess
debug ccsip info
debug ccsip feature sip-profile
```

SIP Copylist の例のデバッグ出力

Ingress from CUCM

Received:

INVITE sip:1001@10.50.228.61:5060 SIP/2.0

Via: SIP/2.0/TCP 10.50.244.3:5060;branch=z9hG4bKaad21bc3ae7e

From: "5001" <sip:5001@10.50.244.3>;tag=100442~cdf43-5020-4e79-a10b-99d406971010-36470319

Contact: <sip:5001@10.50.244.3:5060;transport=tcp>

Copylist Details

00440: Mar 8 18:59:49.796: //-1/xxxxxxxxxxxx/SIP/Info/info/64/sip_profiles_application_peer_copy_patte

000441: Mar 8 18:59:49.797: //187/D6138E000000/SIP/Info/info/64/sip_profiles_application_peer_copy_pat

000442: Mar 8 18:59:49.797: //-1/xxxxxxxxxxxx/SIP/Info/info/64/sip_profiles_prefix_slash_in_copy_var_v

000443: Mar 8 18:59:49.797: //187/D6138E000000/SIP/Info/info/64/sip_profiles_application_peer_copy_pat

000444: Mar 8 18:59:49.797: //-1/xxxxxxxxxxxx/SIP/Info/info/64/sip_profiles_application_modify_remove_

000445: Mar 8 18:59:49.797: //187/D6138E000000/SIP/Info/info/64/sip_profiles_check_and_get_variables_i

000446: Mar 8 18:59:49.797: //187/D6138E000000/SIP/Info/info/64/sip_profiles_check_and_get_variables_i

000448: Mar 8 18:59:49.797: //187/D6138E000000/SIP/Info/info/64/sip_profiles_check_and_get_variables_i

000449: Mar 8 18:59:49.797: //-1/xxxxxxxxxxxx/SIP/Info/info/64/sip_profiles_app_modify_header: Passing

000450: Mar 8 18:59:49.798: //-1/xxxxxxxxxxxx/SIP/Info/info/64/sip_profiles_application_modify_remove_

000451: Mar 8 18:59:49.798: //187/D6138E000000/SIP/Msg/ccsipDisplayMsg:

Egress from CUBE

Sent:

INVITE sip:1001@14.50.228.63:5060 SIP/2.0

Via: SIP/2.0/UDP 10.50.228.61:5060;branch=z9hG4bK3C7CD

Remote-Party-ID: "5001" <sip:5001@10.50.228.61>;party=calling;screen=yes;privacy=off

From: "5001" <sip:5001@10.50.228.61>;tag=34C458-D6

Contact: <sip:5001@168.117.64.94>

特記事項

プロトコル シグナリングとメディア バインド

すべてのシグナリング プロトコルで、管理者はシグナリングを特定のインターフェイスにバインドすることができます。デフォルトでは、スタティックに定義されたバインディングがないゲートウェイは、パケットが通過する物理インターフェイスからコールのシグナリングを発信します。ダイヤルピアのバインディングでは、パケットは指定されたインターフェイスからのソースへ

ッダー、メッセージング、およびパケットを特徴としますが、実際のパケットは引き続き物理インターフェイスを介してルーティングされます。ダイヤルピア バインドは、Session Initiation Protocol (SIP) を含む音声クラス テナントおよびグローバル音声サービス VoIP バインドに常に優先します。

管理者は、ループバックにシグナリングを何回もバインドします。これは論理インターフェイスであるため、このインターフェイスを通過するパケットはありません。パケットキャプチャを実行するには、物理インターフェイスでキャプチャを実行する必要があります。コマンドshow ip cef <remote-ip>では、設定が仮想インターフェイスにバインドされていても、パケットが宛先/リモートIPにルーティングするために利用する物理インターフェイスが表示されます。

メディアおよびシグナリングのバインドは、必ずしも同じ IP である必要はありません。管理者がCUCMとの間でシグナリングを行うために特定のインターフェイスにバインドする必要があるが、電話機とゲートウェイ間の音声/メディアは別のインターフェイスにバインドする必要がある場合。

設定例

この例では、ループバック 1 にバインドされ、CUCM からコールを受信するダイヤルピアを示します。

メディアおよびシグナリング (コントロール) はループバック 1 にバインドされていますが、show ip cef コマンドは、CUCM に送信されるすべてのパケットが物理インターフェイス GigabitEthernet0/0/1 から発信されることを示しています。

```
!  
dial-peer voice 2 voip  
  description "Incoming call from CUCM"  
  session protocol sipv2  
  incoming called-number .  
  voice-class sip bind control source-interface Loopback1  
  voice-class sip bind media source-interface Loopback1  
!
```

レイヤ7アプリケーションバインディングの操作順序

1. 一致する着信/発信ダイヤルピアのbind文に従います。
2. 一致する着信/発信ダイヤルピアに割り当てられた音声クラステナントの下のバインドに従います。
3. グローバルバインディングステートメントに従う
4. 物理層3インターフェイスに基づくと、パケットはルーティングテーブルに基づいて送信されます。

SIPバインディングコマンド

<#root>

```
! Per Dial-peer

!
dial-peer voice 1 voip
  voice-class sip bind control source-interface <interface>
  voice-class sip bind media source-interface <interface>
!

! Global Binding

!
voice service voip
  sip
    bind control source-interface <interface>
    bind media source-interface <interface>
!
```

MGCPバインディングコマンド

```
!
mgcp bind control source-interface <interface>
mgcp bind media source-interface <interface>
!
```

SCCPバインディングコマンド

```
!
sccp local <interface>
!
sccp ccm group <number>
  bind interface <interface>
!
```

H323バインディングコマンド

```
<#root>

!
interface <interface>
!

! Media Bind Command:

h323-gateway voip interface
!

! Signaling Bind Command:

h323-gateway voip bind srcaddr <a.b.c.d>
```

!

DNS および VoIP ダイアルピア

VOIPを使用したDNSは、他のDNSソリューションと同様に採用されています。一般的な設定では、`session target dns:FQDN.com`を使用します。

Cisco ゲートウェイは、`no ip domain lookup` がゲートウェイにグローバルに設定されている場合にも、DNS 解決を実行します。これは、DNSを無効にしても、VoIPダイアルピアは引き続きDNSエントリを解決することを意味します。ただし、最近Cisco IOS XE 3.16Sでは、Cisco IOS XEプラットフォーム内の全体的なDNS機能に変更が加えられました。

この変更後、セッションターゲット`dns:FQDN.com`で設定されたダイアルピアは、`no ip domain lookup`でDNSが無効になるという事実に従うようになります。

この問題を避けるために、DNSを使用する場合は、コマンド「`ip domain lookup`」が設定されていることを常に確認することをお勧めします。

発信SIP接続では、CUBEはDNS解決のためにこの順序の操作を実行します。

1. SRVクエリルックアップ
2. レコード検索
3. AAAAレコードの検索

SRVの作成方法、またはSRVをスキップしてセッションターゲットでAレコードクエリを実行する方法の詳細については、『[Cisco Unified Border Elementコンフィギュレーションガイド - Cisco IOS XE 17.6以降](#)』を参照してください。

メッセージに応答するためにIOSゲートウェイがヘッダーを解決する必要がある着信SIP接続では、ゲートウェイはDNS解決にこの操作順序を使用できます

1. レコード検索
2. AAAAレコードの検索

Cisco IOS XE 17.9.1では、CUBEはオプションキープアライブメカニズムを使用してDNSセッションターゲットの到達可能性を確認できます。詳細なドキュメントを参照してください。

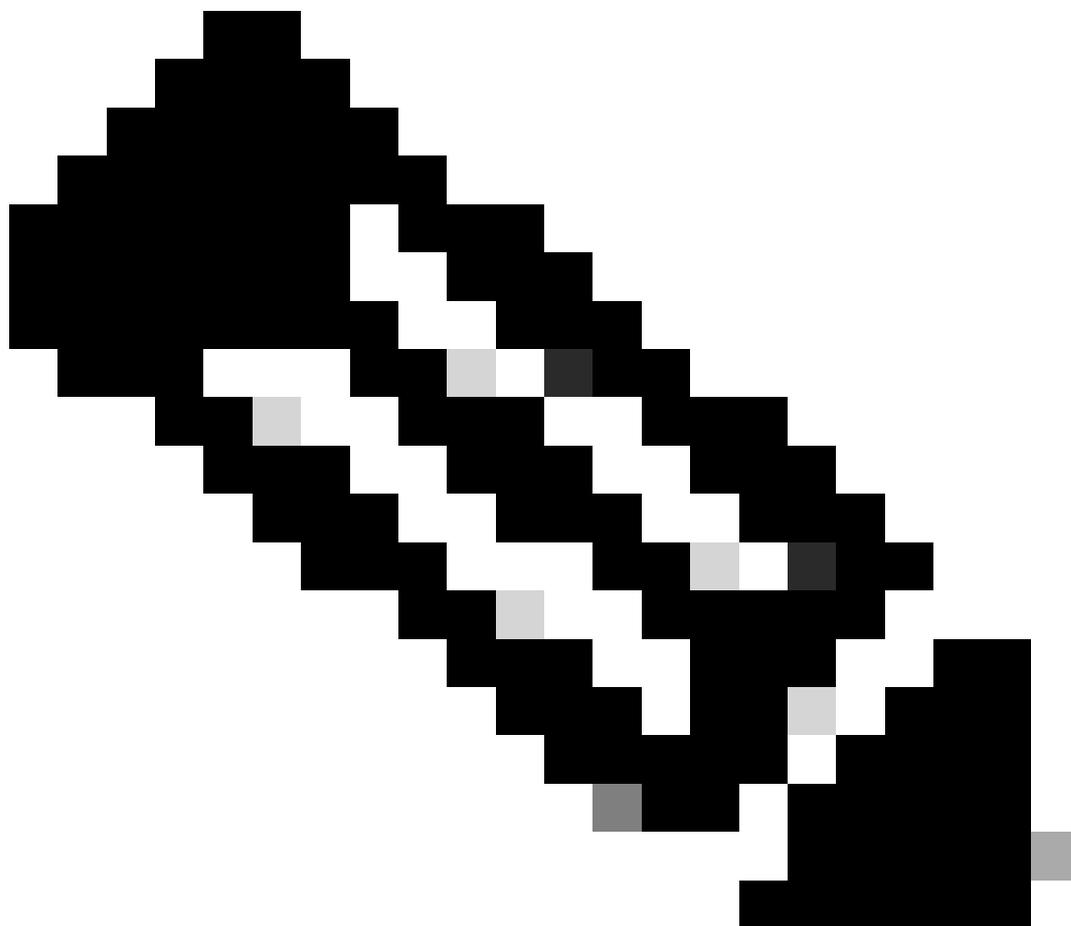
[Cisco Unified Border Elementコンフィギュレーションガイド - Cisco IOS XE 17.6以降](#)

IOS DNSの設定例

```
ip host _sip._udp.cucmgroup.lab.local srv 1 50 5060 cucm1.lab.local
ip host _sip._udp.cucmgroup.lab.local srv 1 50 5060 cucm2.lab.local
ip host _sip._udp.cucmgroup.lab.local srv 1 50 5060 cucm3.lab.local
```

```
ip host cucm1.lab.local 10.0.0.1
ip host cucm2.lab.local 10.0.0.2
ip host cucm3.lab.local 10.0.0.3
```

```
ip domain name lab.local
ip name-server 8.8.8.8
```



注: Cisco IOS XEのDNS SRVサポートは、15.6(1)S / 3.17.00.S以降でサポートされています。

DNSのデバッグと検証コマンド

```
<#root>
```

```
show host
clear host all *
!
debug ip dns view
debug ip domain
debug ccsip info
debug ccsip error
```

DNSテスト3.15S以降

<#root>

Domain Name Verification

```
Gateway# sh run | s lookup
no ip domain lookup
```

Checking the host table for no entry

```
Gateway# show host
Name lookup view: Global
Default domain is cisco.com
Name/address lookup uses static mappings
```

```
Codes: UN - unknown, EX - expired, OK - OK, ?? - revalidate
       temp - temporary, perm - permanent
       NA - Not Applicable None - Not defined
```

Host	Port	Flags	Age	Type	Address(es)
------	------	-------	-----	------	-------------

Verification of no PING on a FQDN

```
Gateway# ping cucm.cisco.com
Translating "cucm.cisco.com"
% Unrecognized host or address, or protocol not running.
```

Made a test call here

Checking logs to see if it worked

```
Gateway# sh log | s INVITE sip:
INVITE sip:9001@14.50.228.70:5060 SIP/2.0
INVITE sip:5001@cucm.cisco.com:5060 SIP/2.0
```

Host Table now has an entry

```
Gateway# sh host
Name lookup view: Global
Default domain is cisco.com
Name/address lookup uses static mappings
```

```
Codes: UN - unknown, EX - expired, OK - OK, ?? - revalidate
       temp - temporary, perm - permanent
       NA - Not Applicable None - Not defined
```

Host	Port	Flags	Age	Type	Address(es)
cucm.cisco.com		None (temp, OK)	0	IP	10.50.244.2

CCSIP All output showing a proper DNS Query for the FQDN on the dial-peer.

```
001338: Mar  9 15:29:07.437: //-1/xxxxxxxxxxxx/SIP/Info/info/1024/httpish_msg_free: Freed msg=0x7FE9873
001339: Mar  9 15:29:07.437: //-1/xxxxxxxxxxxx/SIP/Info/notify/8192/sip_dns_type_srv_query: TYPE SRV qu
001340: Mar  9 15:29:07.438: //-1/xxxxxxxxxxxx/SIP/Info/info/8192/sip_dns_type_a_aaaa_query: DNS query
001341: Mar  9 15:29:07.441: //-1/xxxxxxxxxxxx/SIP/Info/notify/8192/sip_dns_type_a_query: TYPE A query
001342: Mar  9 15:29:07.441: //-1/xxxxxxxxxxxx/SIP/Info/info/8192/sip_dns_type_a_query: ttl for A recor
```

```
001343: Mar 9 15:29:07.441: //-1/xxxxxxxxxxxx/SIP/Info/info/8192/sip_dns_type_a_aaaa_query: IP Address
001344: Mar 9 15:29:07.441: //-1/xxxxxxxxxxxx/SIP/Info/info/8192/sip_dns_type_a_aaaa_query: 10.50.244.
```

DNSテスト3.16S以降。

<#root>

Checking the command is present

```
Gateway# sh run | s lookup
no ip domain lookup
```

Verifying the gateway cannot ping a FQDN

```
Gateway# ping cucm.cisco.com
% Unrecognized host or address, or protocol not running.
```

Checking the Host Table for entries

```
Gateway# sh host
Default domain is cisco.com
Name servers are 10.50.244.52
NAME TTL CLASS TYPE DATA/ADDRESS
-----
```

Made a test call here

CCSIP All Outbound showing the failed call

```
000974: *Mar 9 15:53:01.222: //-1/xxxxxxxxxxxx/SIP/Info/info/1024/httpish_msg_free: Freed msg=0x7FF31D
000975: *Mar 9 15:53:01.222: //-1/xxxxxxxxxxxx/SIP/Info/notify/8192/sip_dns_type_srv_query: TYPE SRV q
000976: *Mar 9 15:53:01.224: //-1/xxxxxxxxxxxx/SIP/Info/info/8192/sip_dns_type_a_aaaa_query: DNS query
000977: *Mar 9 15:53:01.225: //-1/xxxxxxxxxxxx/SIP/Error/sip_dns_type_a_query:
TYPE A query failed for cucm.cisco.com
000978: *Mar 9 15:53:01.225: //-1/xxxxxxxxxxxx/SIP/Error/_send_dns_fail:
DNS Query for cucm.cisco.com failed
000984: *Mar 9 20:53:01.225: %VOICE_IEC-3-GW: SIP: Internal Error (DNS query fail): IEC=10.1.128.7.47.
```

最大接続と帯域幅

デフォルトでは、VoIPおよびPOTSダイヤルピアは無制限の接続（コール）と帯域幅（VoIPダイヤルピアのみ）を許可します。コール数または使用できる帯域幅に制限があるトランクでは、max-connコマンドまたはmax-bandwidthコマンドを使用すると便利です。max-connはCisco IOS 11.3(1)Tで追加され、すべてのCisco IOS XEバージョンに存在します。max-bandwidthは15.2(2)TとIOS-XE 3.7Sで追加されました。

設定例：

ここでは、「max-conn 30」を使用してダイヤルピアを1～30に制限するようにゲートウェイに指示します。

ダイヤルピア2では、割り当てられた制限を超えないように、そのダイヤルピアの帯域幅を制限

しています。

```
!  
dial-peer voice 1 voip  
  description ITSP SIP Trunk - 30 Max Calls!  
  session protocol sipv2  
  sess target ipv4:10.10.10.10  
  destination-pattern 8675309$  
  max-conn 30  
!  
dial-peer voice 2 voip  
  description SIP Trunk with Bandwidth Restrictions!  
  session protocol sipv2  
  sess target ipv4:10.10.10.10  
  destination-pattern 123456789$  
  max-bandwidth 400  
!
```

max-connのしきい値を超えたときのサンプルエラー。

```
000308: Oct  5 19:01:02.603: %CALL_CONTROL-6-MAX_CONNECTIONS: Maximum number of connections reached for  
000309: Oct  5 19:01:02.603: %VOICE_IEC-3-GW: CCAPI: Internal Error (Dial-peer connections exceeded): I  
000310: Oct  5 19:01:02.604: %SIP-3-MAXCONNCAC: Call rejected due to CAC based on maximum number of con  
000311: Oct  5 19:01:02.604: //17084/86B070800000/SIP/Msg/ccsipDisplayMsg:  
Sent:  
SIP/2.0 503 Service Unavailable  
Via: SIP/2.0/TCP 10.50.244.62:5060;branch=z9hG4bKb78c35aa21b0  
From: <sip:9001@10.50.244.62>;tag=72531~2e8ca155-3f0b-4f07-a1b2-b14ef77ceb7f-26250846  
To: <sip:1234@10.50.245.70>;tag=3E19564D-1684  
Date: Thu, 05 Oct 2017 19:01:02 GMT  
Call-ID: 86b07080-9d61816e-b762-3ef4320e@10.50.244.62  
CSeq: 101 INVITE  
Allow-Events: telephone-event  
Warning: 399 10.50.245.70 "Maximum Number of Connections reached for dial-peer 1"  
Server: Cisco-SIPGateway/IOS-15.4.3.S4  
Content-Length: 0
```

ダイヤルイン方式 (DID)

1 段階ダイヤリング

POTSダイヤルピアでダイヤルイン方式を有効にすると、着信メッセージングにコールのルーティングに必要なすべてのディジットを含めることができます。Ciscoゲートウェイは、後続のディジット収集を実行できません。ルータやゲートウェイが発信ダイヤルピアを検索する際、デバイスは着信ダイヤルストリング全体を使用します。この照合は、デフォルトでは可変長です。DID

定義によりすべてのディジットが受信されているため、この照合はディジット単位では行われません。これは、POTSダイヤルピアのデフォルト設定です。

詳細なドキュメント：[IOS音声デジタル\(T1/E1\)を装備したインターフェイスにおけるダイヤルイン方式\(DID\)について](#)

設定例

```
!  
dial-peer voice 1 pots  
  incoming called-number 8675309  
  voice-port 0/0/0  
  direct-inward-dial  
!
```

2 段階ダイヤリング

着信POTSダイヤルピアがno direct-inward-dialを使用して設定されている場合、ルータまたはゲートウェイはディジット収集モードに入ります（ディジットはインバンドで収集されます）。発信ダイヤルピア照合は、ディジット単位で行われます。ルータやゲートウェイは、各ディジットを受信するたびにダイヤルピアとの一致をチェックし、完全に一致するとコールをルーティングします。

設定例

```
!  
dial-peer voice 1 pots  
  incoming called-number 8675309  
  voice-port 0/0/0  
  no direct-inward-dial  
!
```

コールのブロック

各プロトコルは、コールブロックを少し異なる方法で処理します。ほとんどのプロトコルは、ディジットストリングに基づいてブロックするトランスレーションルールの拒否パターンを使用できます。管理者が、通常のディジット操作に着信トランスレーションプロファイルを適用するが、その内部の番号をブロックしない場合、call-block translation-profileコマンドを使用してコールブロックを実装するオプションがあります。

!

```
voice translation-rule 164
  rule 1 reject /8675309/
!
voice translation-profile CALLBLOCK
  translate calling 164
!
dial-peer voice 1 pots
  desc INCOMING VOICE-PORT with BLOCK
  translation-profile incoming ANOTHER
  call-block translation-profile incoming CALLBLOCK
  call-block disconnect-cause incoming invalid-number
  incoming called-number .
  port 0/0/0:23
!
```

```
Gateway#test voice translation-rule 164 8675309
8675309 blocked on rule 1
```

E1 R2 内には、管理者がコレクト コールをブロックする機能があります。これは主にブラジルでの導入で使用されていますが、任意のcas-customグループを使用して設定できます。

次の 2 つのオプションがあります。

1. Telco スイッチから受信したカテゴリ II-8 に基づいて着信コレクト コールをブロックします。これはデフォルト メカニズムであり、すべての Cisco ゲートウェイは設定なしでこれを行います。このブロックの方法には、カテゴリ ベースのマーキングをサポートする新しい Telco スイッチが必要です。この方法を無効にするには、cas-customerグループでコマンド collect-call-enableを使用します。
2. r2-digital E1 R2 cas-custom グループで double-answer 機能グループを使用します。Double-answer設定はカテゴリベースのブロッキングを無効にし、Double-answer機能に置き換えます。これは、最初に着信コールに応答し、1 秒タイマーを開始することで機能します。1秒後、ゲートウェイはCLEAR BWDコマンドの形式でリリースを送信します。電話会社はゲートウェイにCLEAR FWDを送信し、コールを切断できます。タイマーはゲートウェイがCLEAR BWDを送信した後に開始し、このタイマーが時間切れになると、ゲートウェイは別のANSWER信号を送信して、コールがコレクトコールではないと想定し、ここからコールは通常どおりに進みます。このタイマーは、cas-custom グループ内の cc-reanswer-toを使用して設定できます。

カテゴリII-8ブロックメッセージ(debug vpm signal)

<#root>

```
009228: Nov 21 12:02:00.955 GMT: //-1/BF12BE36BAC8/VTSP:(0/0/0:0):-1:1:2/vtsp_report_cas_digit:
  Begin Digit=8, Mode=CC_TONE_R2_MF_BACKWARD_MODE
```

```
009229: Nov 21 12:02:00.955 GMT: htsp_digit_ready_up(0/0/0:0(2)):
```

```
Rx digit='8'
```

```
009230: Nov 21 12:02:00.955 GMT: R2 Incoming Voice(0/0): DSX (E1 0/0/0:1): STATE: R2_IN_CATEGORY R2 Got
```

```
009231: Nov 21 12:02:00.955 GMT: Enter r2_comp_category
```

```
009232: Nov 21 12:02:00.955 GMT:
```

```
R2 Event : 8
```

```

009233: Nov 21 12:02:00.955 GMT:
#####R2_II8 TRUE#####
009234: Nov 21 12:02:00.955 GMT:
##### collect_call_enable = 0
009235: Nov 21 12:02:00.955 GMT:
#####sending B7 #####
009236: Nov 21 12:02:00.955 GMT: r2_reg_generate_digits(0/0/0:0(2)):
Tx digit '7'
009237: Nov 21 12:02:01.055 GMT: //-1/BF12BE36BAC8/VTSP:(0/0/0:0):-1:1:2/vtsp_report_cas_digit:
  End Digit=8, Mode=CC_TONE_R2_MF_BACKWARD_MODE
009238: Nov 21 12:02:01.055 GMT: htsp_digit_ready(0/0/0:0(2)): Rx digit='#'
009239: Nov 21 12:02:01.055 GMT: R2 Incoming Voice(0/0): DSX (E1 0/0/0:1): STATE: R2_IN_CATEGORY R2 Got
009240: Nov 21 12:02:01.055 GMT: Enter r2_comp_category
009241: Nov 21 12:02:01.055 GMT: r2_reg_generate_digits(0/0/0:0(2)): Tx digit '#'
009242: Nov 21 12:02:01.359 GMT: htsp_dsp_message: SEND_SIG_STATUS: state=0x8 timestamp=22365 systime=2
009243: Nov 21 12:02:01.359 GMT: htsp_process_event: [0/0/0:0(2), R2_Q421_IC_WAIT_ANSWER, E_DSP_SIG_100
009244: Nov 21 12:02:01.359 GMT: r2_q421_ic_clr_fwd_idle(0/0/0:0(2)) Rx CLEAR FWD
009245: Nov 21 12:02:01.359 GMT: r2_reg_channel_disconnected(0/0/0:0(2))
009246: Nov 21 12:02:01.359 GMT: R2 Incoming Voice(0/0): DSX (E1 0/0/0:1): STATE: R2_IN_CATEGORY R2 Got
009247: Nov 21 12:02:01.359 GMT: Enter r2_comp_category
009248: Nov 21 12:02:01.359 GMT: htsp_timer - 2000 msec
009249: Nov 21 12:02:01.359 GMT: htsp_process_event: [0/0/0:0(2), R2_Q421_IC_CLR_FWD, E_HTSP_RELEASE_RE
009250: Nov 21 12:02:01.359 GMT: r2_q421_null_release(0/0/0:0(2)) E_HTSP_RELEASE_REQ
009251: Nov 21 12:02:01.359 GMT: r2_reg_process_event: [0/0/0:0(2), R2_REG_COLLECTING, E_R2_REG_DISCONN
009252: Nov 21 12:02:01.359 GMT: r2_reg_disconnect_collect(0/0/0:0(2))
009253: Nov 21 12:02:01.359 GMT: r2_reg_timer_stop(0/0/0:0(2))
009254: Nov 21 12:02:01.711 GMT: htsp_process_event: [0/0/0:0(1), R2_Q421_IC_CLR_FWD, E_HTSP_EVENT_TIME
009255: Nov 21 12:02:01.711 GMT: htsp_timer_stop
009256: Nov 21 12:02:01.711 GMT: r2_q421_clr_fwd_idle(0/0/0:0(1)) Tx IDLEvm_dsp_set_sig_state:[R2 Q.42
009257: Nov 21 12:02:01.711 GMT: r2_reg_channel_disconnected(0/0/0:0(1))
009258: Nov 21 12:02:01.711 GMT: //682206/0C63B263B9C9/VTSP:(0/0/0:0):0:1:1/vtsp_do_call_history:
  Coder Rate=5
009259: Nov 21 12:02:01.711 GMT: r2_reg_process_event: [0/0/0:0(1), R2_REG_IDLE, E_R2_REG_DISCONNECT(91

```

Double-Answer の設定例

```

!
controller e1 0/0/0
  ds0-group 0 timeslots 1-15,17-31 type r2-digital r2-compelled ani
  cas-custom 0
  country brazil
  double-answer
  cc-reanswer-to 3000
!

```

Double-Answer のデバッグ (debug vpm signal)

```
<#root>
```

Answer the call and start a 1 second timer

May 23 09:52:59.180 BR: r2_q421_ic_answer(0/0/0:0(18))

Tx ANSWER

seizure: delay 0 ms,elapsed 12404 msvnm_dsp_set_sig_state:[R2 Q.421 0/0/0:0(18)] set signal state = 0x

May 23 09:52:59.180 BR: r2_reg_channel_connected(0/0/0:0(18))

May 23 09:52:59.180 BR:

htsp_timer - 1000 msec

May 23 09:52:59.180 BR: //23899578/92233E71B421/CCAPI/cc_api_voice_mode_event:

Call Id=23899578

May 23 09:52:59.180 BR: //23899578/92233E71B421/CCAPI/cc_api_voice_mode_event:

Call Entry(Context=0x1E73AD8)

May 23 09:52:59.180 BR: htsp_process_event: [0/0/0:0(18), R2_Q421_IC_DOUBLE_ANS_ANS, E_HTSP_VOICE_CUT_T

May 23 09:52:59.184 BR: //23899578/92233E71B421/CCAPI/cc_process_notify_bridge_done:

Conference Id=0x10AD1, Call Id1=23899578, Call Id2=23899579

May 23 09:52:59.184 BR: r2_reg_process_event: [0/0/0:0(18), R2_REG_WAIT_FOR_CONNECT, E_R2_REG_CONNECT(9

May 23 09:52:59.184 BR: r2_reg_connect(0/0/0:0(18))

One Second Passes and we clear the call and start a 2 second timer

May 23 09:53:00.180 BR: htsp_process_event: [0/0/0:0(18), R2_Q421_IC_DOUBLE_ANS_ANS, E_HTSP_EVENT_TIMER

May 23 09:53:00.180 BR: r2_q421_ic_d_anw_anw_to(0/0/0:0(18)) E_TIMER_EVENT

May 23 09:53:00.180 BR: htsp_timer - 2000 msec

May 23 09:53:00.180 BR: r2_q421_ic_d_anw_anw_to(0/0/0:0(18))

Tx CLEAR BWD

vnm_dsp_set_sig_state:[R2 Q.421 0/0/0:0(18)] set signal state = 0xC

May 23 09:53:00.824 BR: htsp_process_event: [0/0/0:0(18), R2_Q421_IC_DOUBLE_ANS_RLS, E_DSP_SIG_1000]

May 23 09:53:00.824 BR: r2_q421_ic_answer_clr_fwd(0/0/0:0(18))

Rx CLEAR FWD

May 23 09:53:00.824 BR: r2_reg_channel_disconnected(0/0/0:0(18))

May 23 09:53:00.824 BR:

htsp_timer - 2000 msec

May 23 09:53:00.824 BR: r2_reg_process_event: [0/0/0:0(18), R2_REG_CONNECTED, E_R2_REG_DISCONNECT(91)]

May 23 09:53:00.824 BR: r2_reg_disconnect_idle(0/0/0:0(18))

May 23 09:53:00.824 BR: R2 Incoming Voice(0/0): DSX (E1 0/0/0:17): STATE: R2_IN_IDLE R2 Got Event R2_ST

May 23 09:53:00.824 BR: r2_reg_timer_stop(0/0/0:0(18))

2 second passes and the gateway release the call

May 23 09:53:02.824 BR: htsp_process_event: [0/0/0:0(18), R2_Q421_IC_CLR_FWD, E_HTSP_EVENT_TIMER]

May 23 09:53:02.824 BR: htsp_timer_stop

May 23 09:53:02.824 BR: r2_reg_channel_disconnected(0/0/0:0(18))

May 23 09:53:02.824 BR: //23899578/92233E71B421/VTSP:(0/0/0:0):17:1:1/vtsp_cc_call_disconnected:

Cause Value=16

May 23 09:53:02.824 BR: //23899578/92233E71B421/CCAPI/cc_api_call_disconnected:

Cause Value=16, Interface=0xB41CEBC, Call Id=23899578

ISDN オーバーラップ受信

ISDNインターフェイスでisdn overlap-receivingコマンドが設定されている場合は、着信ダイヤル

ピア照合に対する影響があります。ISDNレイヤでディジットが1つ受信されるたびに、照合のためにダイヤルピアがチェックされます。完全一致が見つかり、残りのディジットを待たないで、即時に（この場合はセッションアプリケーションに対して）コールがルーティングされます。Tターミネータを使用して、このディジット単位の照合を中断し、すべてのディジットが受信されるまでルータやゲートウェイを強制的に待機させることができます。「T」はISDNレベルでのT302ディジット間タイマーを指し、ISDNインターフェイスに関連付けられたシリアルインターフェイスで設定できます。ISDNは、Q.931情報メッセージでのSending Complete Information Element (IE)の設定など、ディジットの終了を示すその他のメカニズムも提供しています。

空の着信者番号

ダイヤルピアがincoming called-number Tを使用して設定されている場合、表示される警告メッセージが表示されます。

出力例

```
Gateway(config)# dial-peer voice 1 pots
Gateway(config-dial-peer)# incoming called-number T
Warning: Pattern T defines a match with zero or more digits and hence could
match with an empty number. If this is not the desired behaviour please
configure pattern .T instead to match on one or more digits
```

空の着信者番号による着信ダイヤルピア照合に関する特記事項

- 「Null」の着信番号は、音声ポートや場合によってはanswer-addressと比較して適格性が低いと見なされます。したがって、NULLの着信者番号に基づく照合は、answer-addressまたはport-numberのいずれかに基づく照合が存在しない場合にのみ行われます。
- オーバーラップダイヤルの場合、タイムアウトが発生していないため、「null」の着信番号は「incoming called-number T」と一致しません。
- 「Null」の着信番号は、ENBLOCKの場合にのみ「incoming called-number T」と一致し、answer-addressとポート番号のいずれによっても一致しません。管理者がincoming called-number Tを設定するときに表示される警告は、この特定のケースを示しています。

制限クラス

制限クラス(COR)は、Ciscoゲートウェイでコールを制限する方法です。CORは、よくロックとキーメカニズムと呼ばれます。ロックは、発信CORリストとともにダイヤルピアに割り当てられます。キーは、着信CORリストとともにダイヤルピアに割り当てられます。CORリストが適用されると、使用可能な発信ダイヤルピアはキーがロックを解除できる発信ダイヤルピアになります。このフィルタリングは、残りの発信ダイヤルピア照合方法がチェックされる前に発生します。

制限クラスに関する2つの重要なルール：

1. 発信CORリストが適用されていない場合、コールは常にルーティングされます。

2. 着信CORリストがない場合、コールは常にルーティングされます。

Class of Restriction(COR)、Logical Partitioning Class of Restriction(LPCOR)、およびLPCORと Forced Authorization Codes(FAC)の設定については、このドキュメントでは説明しませんが、これらのドキュメントについては参照可能です。

COR	Class of Restrictions (COR) の設定
CME による LPCOR	LPCOR を伴う CME の設定例
CME および FAC による LPCOR	Cisco Unified Communications Manager Express システム アドミニストレータ ガイド

Cisco Unified Communications Manager Express (CUCME) ダイアルピア

CME は ephone および音声レジスタ プール用のシステム ダイアルピアを作成します。これらは、実行中の設定では表示できません。CMEダイアルピアを変更するには、実際のephoneまたは音声レジスタプールで変更を行う必要があります。show dial-peer voice summary出力を表示すると、2000で始まるダイアルピアはSCCP ephoneで、4000で始まるダイアルピアはSIP音声レジスタプールです。このダイアルピアは、CME 登録電話からのコールの発信ダイアルピア、およびCME 登録電話へのコールのデバッグの発信ダイアルピアとして表示されます。

CMEでのshow dial-peer voice summaryの出力例。

```
Gateway# show dial-peer voice sum | s 2000|4000
20001 pots up up 1001$ 0 50/0/1
20002 pots up up 4001$ 0 50/0/2
20003 pots up up 4002$ 0 50/0/3
20004 pots up up 7001$ 0 50/0/4
20005 pots up up 3009$ 0 50/0/5
20006 pots up up 8810....$ 0 50/0/10
20007 pots up up 8811....$ 0 50/0/11
40001 voip up up 14085151111$ 0 syst ipv4:14.50.214.67:50
40002 voip up up 19725252222$ 0 syst ipv4:14.50.214.67:50
40003 voip up up 85225353333$ 0 syst ipv4:14.50.214.67:50
40004 voip up up 442084445555$ 0 syst ipv4:14.50.214.67:50
40005 voip up up 911$ 0 syst ipv4:14.50.214.67:50
40006 voip up up 18005550100$ 0 syst ipv4:14.50.214.67:50
40008 voip up up 2001$ 0 syst ipv4:14.50.214.51:50
```

SIP CMEでのshow voice register dial-peersの出力例。

```
Gateway# show voice register dial-peers
Dial-peers for Pool 2:
```

```
dial-peer voice 40006 voip
destination-pattern 14085151111$
session target ipv4:14.50.214.67:5060
session protocol sipv2
dtmf-relay rtp-nte
digit collect kpm1
codec g711ulaw bytes 160
no vad
call-fwd-all 8888
after-hours-exempt FALSE
```

```
dial-peer voice 40005 voip
destination-pattern 19725252222$
session target ipv4:14.50.214.67:5060
session protocol sipv2
dtmf-relay rtp-nte
digit collect kpm1
codec g711ulaw bytes 160
no vad
after-hours-exempt FALSE
```

MGCP および SCCP とダイヤルピア

MGCP と SCCP は、ダイヤルピアについて独自のルールに従います。これらが使用する唯一の概念は、コール用の目的の音声ポートを設定する必要があるということです。これ以外は、STCAPP と MGCPAPP プロセスによって処理されます。これらのダイヤルピアの設定を調べる際には、service mgcpapp コマンドまたは service stcapp コマンドのいずれかを使用します。これらは、選択したアプリケーションのダイヤルピアを有効にし、アプリケーションに処理できるダイヤルピアを指示します。

これらのプロトコルをデバッグする場合、出力には着信ダイヤルピアの一致は表示されません。これは常にダイヤルピア0として表示できます。存在しないからです。アプリケーションを処理するコール エージェントはコールを送信するポートをすでに選択しており、ゲートウェイはコールのそのレグを制御できないため、ダイヤルピア照合は意味がありません。ただし、発信ダイヤルピアの一致を確認できます。最終的にプロセスを処理するコール エージェントがコールのその側も制御するため、これは単に表示するためだけのものです。

ダイヤルピアは、選択したアプリケーションに制御する物理音声ポートのみを指示します。この大部分は外部コールエージェントとゲートウェイによって制御されるため、ゲートウェイは指示されたとおりに動作します。ここでは、このセクションの基本的な操作方法を省略し、開始するためのいくつかの設定を示します。

サンプルMGCP設定[CUCM自動設定による*]

```
!
mgcp call-agent 10.10.10.10
mgcp
!
ccm-manager mgcp [codec-all]
ccm-manager config server 10.10.10.10
ccm-manager config
```

```
ccm-manger redundant-host 10.10.10.20
!
voice-port 0/0/0
  description The MGCP port to register
  no shut
!
dial-peer voice 1 pots
  description Defining the Port for the MGCP application
  service mgcpapp
  port 0/0/0
!
hostname myrouter
ip domain name cisco.com
ip name server 10.10.10.30
!
ip tftp source-interface gig0/0/0
!
```

詳細なMGCPドキュメント : [Cisco Unified Communications Managerおよび相互運用性設定ガイド、Cisco IOSリリース15M&T](#)

サンプル SCCP/STCAPP 設定 [CUCM 自動設定による*]

```
!
stcapp ccm-group 1
stcapp
!
sccp local gig0/0/0
sccp ccm 10.10.10.10 id 1 priority 1 version 7.0+
sccp ccm 10.10.10.20 id 1 priority 2 version 7.0+
sccp
!
sccp ccm group 1
  bind interface gig0/0/0
  associate ccm 1 priority 1
  associate ccm 2 priority 2
!
ccm-manager config server 10.10.10.10
ccm-manager sccp local gig0/0/0
ccm-manager sccp
!
voice-port 0/0/0
  description The SCCP port to register
  no shut
!
dial-peer voice 1 pots
  description Defining the Port for the SCCP application
  service stcapp
  port 0/0/0
!
ip tftp source-interface gig0/0/0
!
```

管理者がCUCMによるゲートウェイの設定を希望しない場合は、単純にccm-managerコマンドを

削除します。ダイヤルピアの設定は、概念がどのように機能するかを明確にするために含まれています。ccm-manager設定が存在する場合、CUCMはCUCMのポート設定に基づいてこれらのダイヤルピアを作成するため、ダイヤルピアを実際に定義する必要はありません。CUCMは通常、999で始まり、その後3つの数字が続くダイヤルピアを作成します。

ダイヤルピアを使用したSIP DSAPP

SIP DSAPPは、Cisco IOS XE 16.12.1+およびCUCM 12.5.1SU+で追加されました

この機能を使用すると、FXSなどのアナログ音声ポートをCUCMに登録し、管理することができます。ダイヤルピアは引き続き正常に照合されるため、DSAPPでのコールルーティングはMGCPまたはSCCPとは少し異なります。つまり、ゲートウェイはFXSポートからデジタルを収集し、VoIPダイヤルピアでダイヤルピアルックアップを実行できます。一致が見つかると、INVITEがCUCMエンブロックに送信され、CUCMがさらに番号分析を実行します。

サンプルSIP DSAPP設定[CUCM自動設定による*] | IOS-XE 16.12.1+およびCUCM 12.5.1SU+

```
!  
dsapp line  
!  
voice service voip  
  sip  
    bind control source-interface GigabitEthernet0/0/0  
    bind media source-interface GigabitEthernet0/0/0  
    session transport tcp  
!  
application  
  service dsapp  
    param dialpeer 777  
!  
global  
  service default dsapp  
!  
ccm-manager config server 10.10.10.10  
ccm-manager sipana auto-config local GigabitEthernet0/0/0  
!  
dial-peer voice 777 voip  
  destination-pattern 9T  
  session protocol sipv2  
  session target ipv4:10.10.10.10  
  session transport tcp  
  incoming called-number .  
  voice-class sip extension gw-ana  
  voice-class sip bind control source-interface GigabitEthernet0/0/0  
  dtmf-relay rtp-nte  
  codec g711ulaw  
!  
dial-peer voice 19990100 pots  
  service dsapp  
  destination-pattern 7776  
  voice-class sip extension gw-ana  
  port 0/1/0  
!  
sip-ua  
  registrar ipv4:10.10.10.10 expires 3600 tcp
```

!

詳細なSIP DSAPPドキュメント：[Cisco VG450音声ゲートウェイソフトウェアコンフィギュレーションガイド](#)

コールルーティングのトラブルシューティングと確認

詳細については、このドキュメントを参照してください。

[Unified Border Element\(CUBE\)および時分割多重\(TDM\)ゲートウェイのデバッグ収集の設定](#)

翻訳について

シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。