



## SBC multi-VRF の実装

カスタマー エッジ (CE) デバイス (つまり、顧客宅内ルータ) で SBC の multi-VRF (VPN ルーティングおよび転送) サポート機能を使用すると、プロバイダー エッジ (PE) がパケットを相互に再配布する場合にループの防止に必要な、PE のチェックを抑制する機能を提供します。マルチ VRF では、1 台だけのルータを使用して、通常であれば複数のルータが実行するタスクを完了できます。マルチプロトコル ラベル スイッチング (MPLS) およびボーダー ゲートウェイ プロトコル (BGP) を備える必要がないネットワーク上で実行できます。



(注)

この章で使用するコマンドの詳細な説明については、『*Cisco IOS XR Session Border Controller Command Reference*』を参照してください。この章で言及する他のコマンドについては、コマンドリファレンス マスター インデックス (オンライン検索) を使用して、該当するマニュアルを参照してください。

### SBC multi-VRF の実装機能の履歴

| リリース       | 変更内容  |
|------------|---|
| リリース 3.3.0 | この機能は、Cisco XR 12000 シリーズ ルータで導入されました。  |
| リリース 3.4.0 | 変更なし。   |
| リリース 3.4.1 | Cisco XR 12000 シリーズ ルータで、メディア バイパスが異なる隣接間で発生する可能性があります。 <b>media-bypass</b> コマンドが廃止され、 <b>media-bypass-forbid</b> コマンドが追加されました。 |
| リリース 3.5.0 | 変更なし。   |

## 内容

- 「[Multi-VRF の実装の前提条件](#)」 (P.308)
- 「[マルチ VRF の実装の概要](#)」 (P.308)
- 「[Multi-VRF の実装方法](#)」 (P.309)
- 「[マルチ VRF 実装の設定例](#)」 (P.318)
- 「[その他の関連資料](#)」 (P.322)
- 「[関連コマンドのまとめ](#)」 (P.324)

## Multi-VRF の実装の前提条件

次に、SBC multi-VRF を実装するための前提条件を示します。

- 使用される SBC コマンドの適切なタスク ID を含むタスク グループに関連付けられているユーザグループに属している必要があります。ユーザ グループおよびタスク ID の詳細については、『Cisco IOS XR System Security Configuration Guide』の「Configuring AAA Services on Cisco IOS XR Software」モジュールを参照してください。
- SBC ソフトウェアのパッケージ インストール エンベロップ (PIE) をインストールしてアクティブにする必要があります。  
PIE のインストールに関する詳細については、『Cisco IOS XR Getting Started Guide』の「Upgrading and Managing Cisco IOS XR Software」モジュールを参照してください。
- multi-VRF を実装する前に、SBC を作成しておく必要があります。手順については、「SBC 設定の前提条件」モジュールを参照してください。

## マルチ VRF の実装の概要

カスタマー エッジ (CE) デバイス (つまり、顧客宅内ルータ) で SBC の multi-VRF (VPN ルーティングおよび転送) サポート機能を使用すると、プロバイダー エッジ (PE) がパケットを相互に再配布する場合にループの防止に必要な、PE のチェックを抑制する機能を提供します。マルチ VRF では、1 台だけのルータを使用して、通常であれば複数のルータが実行するタスクを完了できます。マルチプロトコル ラベル スイッチング (MPLS) およびボーダー ゲートウェイ プロトコル (BGP) を備える必要がないネットワーク上で実行できます。

VRF を PE 以外のルータで使用すると、チェックはオフになり、IP プレフィックスへの経路を使用して VRF ルーティング テーブルを正しく入力できます。また、バーチャルプライベート ネットワーク (VPN) 機能は低エンド システムでは完全にサポートされていないため、マルチ VRF が重要になります。マルチ VRF は、インプリケーション アドレス空間によって、1 台のルータ内でルーティング インスタンスを論理的に区切ります。

次に、multi-VRF の機能の概要を説明します。

- 1 台の物理ルータを複数の仮想ルータに分割し、各ルータに独自のインターフェイスのセット、ルーティング テーブル、およびルーティング テーブルを含めることができます。SBC は、カスタマーごとに複数の (重複および独立した) ルーティング テーブル (アドレッシング) をサポートしています。仮想ルーティング コンテキストを使用して、1 台のルータ内のルーティング ドメインを区切ります。
- 複数のルータが必要で、利用可能なルータが 1 台だけの場合に、マルチ VRF を使用できます。
- 1 つの物理インターフェイスを複数の仮想ルータに所属させるには、サブインターフェイス (フレームリレー、ATM、VLAN) を使用します。
- BGP と MPLS は使用されません。
- VRF 間の接続は提供されません (VRF 間での内部エクスポートおよびインポートの場合には、BGP を使用する必要があります)。
- 同じ VPN サイトの 2 つのエンドポイント間にコールを配置すると、SBC はそのエンドポイント間で直接メディアをルーティングして、ネットワーク使用率を削減できます。
- SBC で multi-VRF を実現すると、両方のエンドポイントが同じ VPN にある場合には、メディアバイパスを有効にして、最適化を図ることができます。

リリース 3.4.1 の場合、デフォルトでは同じ VPN 上のすべての隣接でメディア バイパスが有効になっています。 **media-bypass-forbid** コマンド（このコマンドは CAC ポリシー専用の実装されています）を使用すると、メディア バイパスをオフにできます。

## Multi-VRF の実装方法

SBC multi-VRF の実装については、次の項を参照してください。

- [multi-VRF の設定](#)
- [H.323 隣接への VRF の関連付け](#)
- [SIP 隣接と VRF との関連付け](#)
- [DBE への VRF の設定（分散モデル専用）](#)

## multi-VRF の設定

このタスクでは、SBC が統合配置モデルの multi-VRF モードで実行されるルータを設定します。必要に応じて、インターフェイスと、SBC の SVI、隣接、および DBE メディア アドレスとの間の関係をメモします。

### 手順の概要

1. **configure**
2. **interface sbc number**
3. **vrf vrf-name**
4. **ipv4 address address prefix**
5. **service-location preferred-active node-id [preferred-standby node-id]**
6. **sbc service-name**
7. **service-location preferred-active node-id [preferred-standby node-id]**
8. **sbc**
9. **activate**
10. H.323 隣接と VRF を関連付けます（「[H.323 隣接への VRF の関連付け](#)」を参照）。
11. SIP 隣接と VRF を関連付けます（「[SIP 隣接と VRF との関連付け](#)」を参照）。
12. **dbe**
13. **media-address ipv4 ipv4\_address [vrf vrf\_name]**
14. **activate**
15. **end**

## 手順の詳細

|       | コマンドまたはアクション  | 目的   |
|-------|---|--|
| ステップ1 | <b>configure</b><br><br>例：<br>RP/0/0/CPU0:router# configure<br>RP/0/0/CPU0:router(config)#  | グローバル コンフィギュレーション モードをイネーブルにします。   |
| ステップ2 | <b>interface sbc number</b><br><br>例：<br>RP/0/0/CPU0:router(config-if)# interface SBC1  | SBC インターフェイス（必要に応じて作成）のモードを開始します。 <i>number</i> 引数は、1 ~ 2000 の値である必要があります。 |
| ステップ3 | <b>vrf vrf_name</b><br><br>例：<br>RP/0/0/CPU0:router(config-if)# vrf my_vrf1   | 特定のバーチャル プライベート ネットワーク（VPN）に H.323 隣接を関連付けます。                              |
| ステップ4 | <b>ipv4 address address prefix</b><br><br>例：<br>RP/0/0/CPU0:router(config-if)# ipv4 address<br>88.88.101.10 255.255.255.0   | インターフェイスに IPv4 アドレスを割り当てます。  |
| ステップ5 | <b>service-location preferred-active node-id</b><br>[ <b>preferred standby node-id</b> ]<br><br>例：<br>RP/0/0/CPU0:router(config-if)# service-location<br>preferred-active 0/4/CPU0 preferred-standby<br>0/5/CPU0  | サービス カードが SBC 機能をプライマリとして、およびオプションでセカンダリ ロケーションとして実行できるようにします。             |
| ステップ6 | <b>sbc service-name</b><br><br>例：<br>RP/0/0/CPU0:router(config-if)# sbc lite_8  | SBC サービスのモードを開始します。<br><br>• <i>service-name</i> 引数を使用して、サービスの名前を定義します。    |
| ステップ7 | <b>service-location preferred-active node-id</b><br>[ <b>preferred standby node-id</b> ]<br><br>例：<br>RP/0/0/CPU0:router(config-sbc)#<br>service-location preferred-active 0/4/CPU0<br>preferred-standby 0/5/CPU0 | サービス カードが SBC 機能をプライマリとして、およびオプションでセカンダリ ロケーションとして実行できるようにします。             |
| ステップ8 | <b>sbe</b><br><br>例：<br>RP/0/0/CPU0:router(config-sbc)# sbe<br>RP/0/0/CPU0:router(config-sbc-sbe)#  | SBC サービス内で SBE エンティティのモードを開始します。   |

|        | コマンドまたはアクション   | 目的   |
|--------|--|--|
| ステップ9  | <code>activate</code><br><br>例：<br>RP/0/0/CPU0:router(config-sbc-sbe-acc)#<br>activate   | SBC サービスを開始します。  |
| ステップ10 | H.323 隣接 <code>h323my_vrf1</code> を <code>vpn1</code> に関連付けます。<br><br>例：<br>「 <a href="#">H.323 隣接への VRF の関連付け：例</a> 」の項を参照してください。 | H.323 隣接を VPN に関連付けます。「 <a href="#">H.323 隣接への VRF の関連付け</a> 」の項を参照してください。 |
| ステップ11 | SIP 隣接 <code>sipmy_vrf1</code> を <code>vpn1</code> に関連付けます。<br><br>例：<br>「 <a href="#">SIP 隣接と VRF との関連付け：例</a> 」の項を参照してください。      | SIP 隣接を VPN に関連付けます。「 <a href="#">SIP 隣接と VRF との関連付け</a> 」の項を参照してください。     |
| ステップ12 | <code>dbe</code><br><br>例：<br>RP/0/0/CPU0:router(config-sbc-sbe)# <code>dbe</code><br>RP/0/0/CPU0:router(config-sbc-dbe)#          | SBC サービス内で DBE エンティティのモードを開始します。   |
| ステップ13 | <code>media-address ipv4 ipv4_address [vrf vrf_name]</code><br><br>例：<br>RP/0/0/CPU0:router(config-sbc-dbe)#<br>media-address      | DBE メディア アドレス プール内に IPv4 アドレスを作成します。                                       |
| ステップ14 | <code>activate</code><br><br>例：<br>RP/0/0/CPU0:router(config-sbc-sbe-acc)#<br>activate   | SBC サービスを開始します。  |
| ステップ15 | <code>end</code><br><br>例：<br>RP/0/0/CPU0:router(config-sbc-sbe-acc)#<br>activate  | 現在のコンフィギュレーション セッションを終了します。  |

## H.323 隣接への VRF の関連付け

このタスクでは、H.323 隣接を VPN に関連付けます。

### 手順の概要

1. `adjacency h323 adjacency-name`
2. `vrf vrf_name`
3. `signaling-address ipv4 local_signaling_IP_address`

4. **signaling-port** *port\_num*
5. **remote-address ipv4** *remote\_IP\_address/prefix*
6. **signaling-peer** [*gk*] *peer\_address*
7. **signaling-peer-port** *port\_num*
8. **dtmf-relay** *rtp-nte*
9. **account** *account\_name*
10. **media-bypass** (Optional command)
11. **media-bypass-forbid**
12. **attach**

## 手順の詳細

|       | コマンドまたはアクション   | 目的  |
|-------|--|---|
| ステップ1 | <b>adjacency</b> <i>h323 adjacency-name</i><br><br>例:<br>RP/0/0/CPU0:router(config-sbc-sbe)# adjacency<br>h323 h323my_vrfl<br>RP/0/0/CPU0:router(config-sbc-sbe-adj-h323)# | SBE H.323 隣接のモードを開始します。<br><br>• <i>adjacency-name</i> 引数を使用して、サービスの名前を定義します。 |
| ステップ2 | <b>vrf</b> <i>vrf_name</i><br><br>例:<br>RP/0/0/CPU0:router(config-sbc-sbe-adj-h323)#<br>vrf my_vrfl  | 特定のバーチャルプライベートネットワーク (VPN) に H.323 隣接を関連付けます。                                 |
| ステップ3 | <b>signaling-address ipv4</b><br><i>local_signaling_IP_address</i><br><br>例:<br>RP/0/0/CPU0:router(config-sbc-sbe-adj-h323)#<br>signaling-address ipv4 88.88.101.11        | H.323 隣接のローカル IPv4 シグナリング アドレスを指定します。   |
| ステップ4 | <b>signaling-port</b> <i>port_num</i><br><br>例:<br>RP/0/0/CPU0:router(config-sbc-sbe-adj-h323)#<br>signaling-port 1720   | H.323 隣接のローカル シグナリング ポートを指定します。   |
| ステップ5 | <b>remote-address ipv4</b> <i>ipv4_IP_address/prefix</i><br><br>例:<br>RP/0/0/CPU0:router(config-sbc-sbe-adj-h323)#<br>remote-address ipv4 10.10.101.4/32                   | 隣接を通じてコンタクトされる一連のリモート シグナリング ピアを、特定の IP アドレス プレフィックスを持つものに限定します。              |
| ステップ6 | <b>signaling-peer</b> [ <i>gk</i> ] <i>peer_address</i><br><br>例:<br>RP/0/0/CPU0:router(config-sbc-sbe-adj-h323)#<br>signaling-peer gk 10.10.101.4                         | H.323 隣接が使用するリモート シグナリング ピアを指定します。  |

|        | コマンドまたはアクション  | 目的   |
|--------|---|--|
| ステップ7  | <code>signaling-peer-port port_num</code><br><br>例：<br>RP/0/0/CPU0:router (config-sbc-sbe-adj-h323) #<br>signaling-peer-port 1720 | H.323 隣接が使用するリモート シグナリング ピア ポートを指定します。   |
| ステップ8  | <code>dtmf-relay [rtp-nte]</code><br><br>例：<br>RP/0/0/CPU0:router (config-sbc-sbe-adj-h323) #<br>dtmf-relay                       | H.323 隣接の DTMF リレーを設定します。<br><br>• <code>rtp-nte</code> を指定すると、RFC 2833 <i>Named Telephone Event (rtp-nte)</i> サポートがイネーブルになります。これを指定しないと、 <code>rtp-nte</code> サポートはディセーブルになります。 |
| ステップ9  | <code>account account_name</code><br><br>例：<br>RP/0/0/CPU0:router (config-sbc-sbe-adj-h323) #<br>account h323-my_vrf1             | SBE 上のアカウントに属するものとして H.323 隣接を定義します。   |
| ステップ10 | <code>media-bypass</code><br><br>例：<br>RP/0/0/CPU0:router (config-sbc-sbe-adj-h323) #<br>media-bypass                             | (任意) メディア トラフィックが DBE をバイパスできるように隣接を設定します。<br><br>このコマンドは任意であり、1 つの隣接でだけ機能します。   |
| ステップ11 | <code>media-bypass-forbid</code><br><br>例：<br>RP/0/0/CPU0:router (config-sbc-sbe-adj-h323) #<br>media-bypass-forbid               | (リリース 3.4.1) メディア トラフィックが DBE をバイパスするのを禁止するように H.323 隣接を設定します。<br><br>設定されていない場合は、この隣接で開始および終了する通話のメディア トラフィックはエンドポイント間を直接フローし、両方の隣接が同じ VPN にある限りは DBE を経由しません。                  |
| ステップ12 | <code>attach</code><br><br>例：<br>RP/0/0/CPU0:router (config-sbc-sbe-adj-h323) #<br>attach   | 隣接をアタッチします。  |

## SIP 隣接と VRF との関連付け

このタスクでは、SIP 隣接を VPN と関連付けます。

### 手順の概要

1. `adjacency sip adjacency-name`
2. `vrf vrf_name`
3. `signaling-address ipv4 local_signaling_IP_address`
4. `signaling-port port_num`
5. `remote-address ipv4 local_signaling_IP_address/prefix`
6. `local-id host name`
7. `signaling-peer [gk] peer_address`

8. `signaling-peer-port port_num`
9. `account account-name`
10. `media-bypass` (任意)
11. `media-bypass-forbid`
12. `attach`

## 手順の詳細

|       | コマンドまたはアクション   | 目的  |
|-------|--|---|
| ステップ1 | <b>adjacency sip adjacency-name</b><br><br><b>例:</b><br>RP/0/0/CPU0:router(config-sbc-sbe)# adjacency sip sipGW<br>RP/0/0/CPU0:router(config-sbc-sbe-adj-sip)# | SBE SIP 隣接のモードを開始します。<br><br><ul style="list-style-type: none"> <li><code>adjacency-name</code> 引数を使用して、サービスの名前を定義します。</li> </ul> |
| ステップ2 | <b>vrf vrf_name</b><br><br><b>例:</b><br>RP/0/0/CPU0:router(config-sbc-sbe-adj-h323)# vrf my_vrf1   | 特定のバーチャル プライベート ネットワーク (VPN) に H.323 隣接を関連付けます。   |
| ステップ3 | <b>signaling-address ipv4 ipv4_IP_address</b><br><br><b>例:</b><br>RP/0/0/CPU0:router(config-sbc-sbe-adj-sip)# signaling-address ipv4 88.88.88.88.101.11        | SIP 隣接のローカル IPv4 シグナリング アドレスを指定します。   |
| ステップ4 | <b>signaling-port port_num</b><br><br><b>例:</b><br>RP/0/0/CPU0:router(config-sbc-sbe-adj-sip)# signaling-port 5060   | SIP 隣接のローカル シグナリング ポートを指定します。   |
| ステップ5 | <b>remote-address ipv4 remote_IP_address/prefix</b><br><br><b>例:</b><br>RP/0/0/CPU0:router(config-sbc-sbe-adj-sip)# remote-address ipv4 10.10.101.4/32         | 隣接を通じてコンタクトされる一連のリモート シグナリング ピアを、特定の IP アドレス プレフィックスを持つものに限定します。  |
| ステップ6 | <b>local-id host address</b><br><br><b>例:</b><br>RP/0/0/CPU0:router(config-sbc-sbe-adj-sip)# local-id host 88.88.101.11  | SIP 隣接のローカル識別名を設定します。   |
| ステップ7 | <b>signaling-peer [gk] peer_address</b><br><br><b>例:</b><br>RP/0/0/CPU0:router(config-sbc-sbe-adj-sip)# signaling-peer 10.10.101.4                             | SIP 隣接が使用するリモート シグナリング ピアを指定します。  |



|         | コマンドまたはアクション  | 目的  |
|---------|---|---|
| ステップ 8  | <b>signaling-peer-port</b> <i>port_num</i><br><br>例：<br>RP/0/0/CPU0:router(config-sbc-sbe-adj-sip)#<br>signaling-peer-port 5060 | SIP 隣接が使用するリモート シグナリング ピア ポートを指定します。  |
| ステップ 9  | <b>account</b> <i>account_name</i><br><br>例：<br>RP/0/0/CPU0:router(config-sbc-sbe-adj-sip)#<br>account sipmy_vrf2               | SIP 隣接を SBE のアカウントに所属しているものとして定義します。  |
| ステップ 10 | <b>media-bypass</b><br><br>例：<br>RP/0/0/CPU0:router(config-sbc-sbe-adj-sip)#<br>media-bypass                                    | (任意) メディア トラフィックが DBE をバイパスできるように隣接を設定します。<br><br>このコマンドは任意であり、1 つの隣接だけで動作します。  |
| ステップ 11 | <b>media-bypass-forbid</b><br><br>例：<br>RP/0/0/CPU0:router(config-sbc-sbe-adj-sip)#<br>media-bypass-forbid                      | (リリース 3.4.1) メディア トラフィックが DBE をバイパスするのを禁止するように SIP 隣接を設定します。<br><br>設定されていない場合は、この隣接で開始および終了する通話のメディア トラフィックはエンドポイント間を直接フローし、両方の隣接が同じ VPN にある限りは DBE を経由しません。 |
| ステップ 12 | <b>attach</b><br><br>例：<br>RP/0/0/CPU0:router(config-sbc-sbe-adj-sip)#<br>attach  | 隣接をアタッチします。   |

## DBE への VRF の設定（分散モデル専用）

このタスクでは、分散モデルで VRF を使用して DBE を設定します。

### 手順の概要

1. **configure**
2. **interface sbc number**
3. **vrf vrf-name**
4. **ipv4 address address prefix**
5. **service-location preferred-active node-id [preferred-standby node-id]**
6. **interface sbc number**
7. **ipv4 address address prefix**
8. **service-location preferred-active node-id [preferred-standby node-id]**
9. **sbc service-name**
10. **service-location preferred-active node-id [preferred-standby node-id]**
11. **dbe**

12. `vdbe [global]`
13. `control-address h248 ipv4 IPv4_H.248_IP_address`
14. `controller h248 controller-index`
15. `transport [udp | tcp]`
16. `remote-address ipv4 remote-address`
17. `attach-controllers`
18. `media-address ipv4 ipv4_address [vrf vrf_name]`
19. `activate`

## 手順の詳細

|       | コマンドまたはアクション  | 目的   |
|-------|---|--|
| ステップ1 | <b>configure</b><br><br><b>例:</b><br>RP/0/0/CPU0:router# configure<br>RP/0/0/CPU0:router(config)#   | グローバル コンフィギュレーション モードをイネーブルにします。   |
| ステップ2 | <b>interface sbcnumber</b><br><br><b>例:</b><br>RP/0/0/CPU0:router(config)# interface SBC1<br>RP/0/0/CPU0:router(config-if)#   | SBC インターフェイス（必要に応じて作成）のモードを開始します。 <i>number</i> 引数は、1 ~ 2000 の値である必要があります。 |
| ステップ3 | <b>vrf vrf_name</b><br><br><b>例:</b><br>RP/0/0/CPU0:router(config-if)# vrf my_vrf1  | 特定のバーチャル プライベート ネットワーク（VPN）に H.323 隣接を関連付けます。                              |
| ステップ4 | <b>ipv4 address address prefix</b><br><br><b>例:</b><br>RP/0/0/CPU0:router(config-if)# ipv4 address<br>88.88.130.10 255.255.255.0  | インターフェイスに IPv4 アドレスを割り当てます。  |
| ステップ5 | <b>service-location preferred-active node-id</b><br><b>[preferred standby node-id]</b><br><br><b>例:</b><br>RP/0/0/CPU0:router(config-if)# service-location<br>preferred-active 0/1/CPU0 | サービス カードが SBC 機能をプライマリとして、およびオプションでセカンダリ ロケーションとして実行できるようにします。             |
| ステップ6 | <b>interface sbc number</b><br><br><b>例:</b><br>RP/0/0/CPU0:router(config-if)# interface SBC2   | SBC インターフェイス（必要に応じて作成）のモードを開始します。 <i>number</i> 引数は、1 ~ 2000 の値である必要があります。 |

|        | コマンドまたはアクション  | 目的   |
|--------|---|--|
| ステップ7  | <b>ipv4 address address prefix</b><br><br>例：<br>RP/0/0/CPU0:router(config-if)# ipv4 address<br>88.88.130.10 255.255.255.0   | インターフェイスに IPv4 アドレスを割り当てます。  |
| ステップ8  | <b>service-location preferred active node-id</b><br><b>[preferred standby node-id]</b><br><br>例：<br>RP/0/0/CPU0:router(config-if)# service-location<br>preferred-active 0/1/CPU0  | サービスカードが SBC 機能をプライマリとして、およびオプションでセカンダリ ロケーションとして実行できるようにします。  |
| ステップ9  | <b>sbc service-name</b><br><br>例：<br>RP/0/0/CPU0:router(config-if)# sbc lite_1<br>RP/0/0/CPU0:router(config-sbc)#   | SBC サービスのモードを開始します。 <ul style="list-style-type: none"> <li><i>service-name</i> 引数を使用して、サービスの名前を定義します。</li> </ul> |
| ステップ10 | <b>service-location preferred active node-id</b><br><b>[preferred standby node-id]</b><br><br>例：<br>RP/0/0/CPU0:router(config-sbc)#<br>service-location preferred-active 0/1/CPU0 | サービスカードが SBC 機能をプライマリとして、およびオプションでセカンダリ ロケーションとして実行できるようにします。  |
| ステップ11 | <b>dbe</b><br><br>例：<br>RP/0/0/CPU0:router(config-sbc)# dbe<br>RP/0/0/CPU0:router(config-sbc-dbe)#  | SBC サービス内で DBE エンティティを設定するモードを開始します。   |
| ステップ12 | <b>vdbe [global]</b><br><br>例：<br>RP/0/0/CPU0:router(config-sbc-dbe)# vdbe<br>RP/0/0/CPU0:router(config-sbc-dbe-vdbe)#  | 仮想 DBE パラメータを設定するモードを開始します。  |
| ステップ13 | <b>control-address h248 ipv4 ipv4_H.248_IP_address</b><br><br>例：<br>RP/0/0/CPU0:router(config-sbc-dbe-vdbe)#<br>control-address h248 ipv4 88.88.130.100                           | 特定の IPv4 H.248 制御アドレスを使用するように DBE を設定します。  |
| ステップ14 | <b>controller h248 controller-index</b><br><br>例：<br>RP/0/0/CPU0:router(config-sbc-dbe-vdbe)#<br>controller h248 1<br>RP/0/0/CPU0:router(config-sbc-dbe-vdbe-h248)#               | DBE の H.248 コントローラを設定するモードを開始します。  |

|         | コマンドまたはアクション   | 目的   |
|---------|--|--|
| ステップ 15 | <code>transport [udp   tcp]</code><br><br>例：<br>RP/0/0/CPU0:router(config-sbc-dbe-vdbe-h248)#<br>transport udp   | H.248 コントローラで、H.248 制御シグナリングに UDP または TCP を使用するように DBE を設定します。 |
| ステップ 16 | <code>remote-address ipv4 remote-address</code><br><br>例：<br>RP/0/0/CPU0:router(config-sbc-dbe-vdbe-h248)#<br>remote-address ipv4 88.88.101.21                 | H.248 コントローラの SBE 上に接続するリモートアドレスを定義します。                        |
| ステップ 17 | <code>attach-controllers</code><br><br>例：<br>RP/0/0/CPU0:router(config-sbc-dbe-vdbe-h248)#<br>attach-controllers<br>RP/0/0/CPU0:router(config-sbc-dbe-vdbe)#   | コントローラに接続する DBE を設定します。  |
| ステップ 18 | <code>media-address ipv4 ipv4_address [vrf vrf_name]</code><br><br>例：<br>RP/0/0/CPU0:router(config-sbc-dbe-vdbe)#<br>media-address ipv4 88.88.130.2 vrf my_vrf | DBE メディア アドレス プールを設定するモードを開始し、VRF で使用するアドレス プールを作成します。         |
| ステップ 19 | <code>activate</code><br><br>例：<br>RP/0/0/CPU0:router(config-sbc-dbe-vdbe)#<br>activate  | SBC サービスを開始します。  |

## マルチ VRF 実装の設定例

ここでは、次の設定例について説明します。

- [マルチ VRF の設定 : 例](#)
- [H.323 隣接への VRF の関連付け : 例](#)
- [SIP 隣接と VRF との関連付け : 例](#)
- [DBE への VRF の設定 \(分散モデル専用\) : 例](#)

## マルチ VRF の設定 : 例

この例では、Service Virtual Interface (SVI; サービス仮想インターフェイス) と隣接を追加して、VPN を関連付ける方法について示します。callgen7 からコールが発信されると、sbc1 はそれを取得し、sbc2 にブリッジングし、callgen8 に渡します。



(注) この設定では、ゲートウェイの隣接および番号分析が削除されています。

1. ラインカード インターフェイスを my\_vrf2 に関連付ける設定をします。

```
cdp
 vrf my_vrf2
  ipv4 address 192.168.229.1 255.255.255.0
  negotiation auto
!
interface GigabitEthernet0/3/0/1
 description vrf-lite 1 to node1
```

2. ラインカード インターフェイスを my\_vrf1 に関連付ける設定をします。

```
cdp
 vrf my_vrf1
  ipv4 address 1.0.4.1 255.255.255.0
  negotiation auto
!
interface SBC1
 vrf my_vrf1
```

3. SVI を my\_vrf1 に関連付ける設定をします。

```
ipv4 address 88.88.101.10 255.255.255.0
 service-location preferred-active 0/4/CPU0 preferred-standby 0/5/CPU0
!
interface SBC2
 vrf my_vrf2
```

4. SVI を my\_vrf2 に関連付ける設定をします。

```
ipv4 address 88.88.129.10 255.255.255.0
 service-location preferred-active 0/4/CPU0 preferred-standby 0/5/CPU0
!
!
sbc lite_8
 service-location preferred-active 0/4/CPU0 preferred-standby 0/5/CPU0
 sbe
  activate
  adjacency h323 h323my_vrf1
  vrf my_vrf1
```

5. H.323 の隣接を my\_vrf1 に関連付ける設定をします。

```
signaling-address ipv4 88.88.101.11
 signaling-port 1720
 remote-address ipv4 10.10.101.4/32
 signaling-peer 10.10.101.4
 signaling-peer-port 1720
 dtmf-relay
 account h323-my_vrf1
 attach
!
adjacency h323 h323my_vrf2
 vrf my_vrf2
```

6. H.323 の隣接を my\_vrf2 に関連付ける設定をします。

```
signaling-address ipv4 88.88.129.11
  signaling-port 1720
  remote-address ipv4 10.10.115.4/32
  signaling-peer 10.10.115.4
  signaling-peer-port 1720
  dtmf-relay
  account h323-my_vrf2
  attach
!
adjacency sip sipmy_vrf1
vrf my_vrf1
```

7. SIP の隣接を my\_vrf1 に関連付ける設定をします。

```
signaling-address ipv4 88.88.101.11
  signaling-port 5060
  remote-address ipv4 10.10.101.4/32
  local-id host 88.88.101.11
  signaling-peer 10.10.101.4
  signaling-peer-port 5060
  account sip-my_vrf1
  attach
!
adjacency sip sipmy_vrf2
vrf my_vrf2
```

8. SIP の隣接を my\_vrf2 に関連付ける設定をします。

```
signaling-address ipv4 88.88.129.11
  signaling-port 5060
  remote-address ipv4 10.10.115.4/32
  local-id host 88.88.129.11
  signaling-peer 10.10.115.4
  signaling-peer-port 5060
  account sip-my_vrf2
  attach
!
!
dbe
media-address ipv4 88.88.101.2 vrf my_vrf1
```

9. メディア アドレスを my\_vrf1 に関連付ける設定をします。

```
media-address ipv4 88.88.129.2 vrf my_vrf2
```

10. メディア アドレスを my\_vrf2 に関連付ける設定をします。

```
activate
!
!
end
```

## H.323 隣接への VRF の関連付け：例

この設定例では、VPN に関連付けられた H.323 隣接を作成します。

```
adjacency h323 h323my_vrf1
  vrf my_vrf1
  signaling-address ipv4 88.88.101.11
  signaling-port 1720
  remote-address ipv4 10.10.101.4/32
  signaling-peer 10.10.101.4
  signaling-peer-port 1720
  dtmf-relay
  account h323-my_vrf1
  attach
```

## SIP 隣接と VRF との関連付け：例

この設定では、VPN に関連付けられた SIP 隣接を作成する例を示します。

```
adjacency sip sipmy_vrf1
  vrf my_vrf1
  signaling-address ipv4 88.88.101.11
  signaling-port 5060
  remote-address ipv4 10.10.101.4/32
  local-id host 88.88.101.11
  signaling-peer 10.10.101.4
  signaling-peer-port 5060
  account sip-my_vrf1
  attach
```

## DBE への VRF の設定（分散モデル専用）：例

次の例では、分散モデルの DBE に VRF を設定する方法を示します。

my\_vrf1 にはメディア アドレスのみ含むことができ、制御アドレス シグナリングはデフォルトの VRF に含める必要があります。統合モデルの SBE および DBE と同様、メディア アドレスは SVI と、my\_vrf1 の物理インターフェイスに関連付けられます。



(注) 制御シグナリング用に 2 つめの SVI がデフォルト VRF が必要になります。

```
interface SBC1
  vrf my_vrf1
  ipv4 address 88.88.130.10 255.255.255.0
  service-location preferred-active 0/1/CPU0
  !
interface SBC2
  ipv4 address 88.88.130.10 255.255.255.0
  service-location preferred-active 0/1/CPU0
  !
```

```

sbclite_1
service-location preferred-active 0/1/CPU0
dbs
vdbs
control-address h248 ipv4 88.88.130.100
controller h248 1
transport udp
remote-address ipv4 88.88.101.21
!
attach-controllers
!
media-address ipv4 88.88.130.2 vrf my_vrf1
activate

```

## その他の関連資料

ここでは、SBC multi-VRF の実装に関する関連資料について説明します。

## 関連資料

| 関連項目  | マニュアル タイトル   |
|---|--|
| Cisco IOS XR マスター コマンド リファレンス               | 『Cisco IOS XR Master Commands List』                        |
| Cisco IOS XR SBC インターフェイス コンフィギュレーション コマンド  | 『Cisco IOS XR Session Border Controller Command Reference』 |
| Cisco IOS XR ソフトウェアを使用するルータを初回に起動し設定するための情報 | 『Cisco IOS XR Getting Started Guide』                       |
| Cisco IOS XR コマンド モード                       | 『Cisco IOS XR Command Mode Reference』                      |

## 標準

| 標準   | タイトル |
|--|------|
| この機能でサポートされる新規の標準または変更された標準はありません。また、既存の標準のサポートは変更されていません。 | —    |



## MIB

| MIB | MIB のリンク   |
|-----|--|
| —   | <p>Cisco IOS XR ソフトウェアを使用して MIB の場所を特定してダウンロードするには、次の URL にある Cisco MIB Locator を使用して、[Cisco Access Products] メニューからプラットフォームを選択します。</p> <p><a href="http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml">http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml</a></p> |

## RFC

| RFC      | タイトル                                       |
|----------|--|
| RFC 2685 | 『Virtual Private Networks Identifier』      |
| RFC 1918 | 『Address Allocation for Private Internets』 |
| RFC 2547 | 『BGP/MPLS VPNs』                            |

## シスコのテクニカル サポート

| 説明  | リンク  |
|---|--|
| <p>シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクなどの、数千ページに及ぶ技術情報が検索可能です。Cisco.com に登録済みのユーザは、このページから詳細情報にアクセスできます。</p> | <p><a href="http://www.cisco.com/en/US/support/index.html">http://www.cisco.com/en/US/support/index.html</a></p> |

## 関連コマンドのまとめ

ここでは、Cisco XR 12000 シリーズ ルータでの SBC multi-VRF の設定に関連するコマンドの一覧をアルファベット順に示します。コマンドの詳細については、『Cisco IOS XR Session Border Controller Command Reference』を参照してください。

| コマンド   | 目的   |
|--|--|
| <code>interface sbc number</code>  | SBC インターフェイス（必要に応じて作成）のモードを開始します。 <i>number</i> 引数は、1 ～ 2000 の値である必要があります。 |
| <code>ipv4 address address prefix</code>   | インターフェイスに IPv4 アドレスを割り当てます。  |
| <code>media-address ipv4 ipv4_address [vrf vrf_name]</code>                        | DBE メディア アドレス プール内に IPv4 アドレスを作成します。                                       |
| <code>service-location preferred active node-id [preferred standby node-id]</code> | サービス カードが SBC 機能をプライマリとして、およびオプションでセカンダリ ロケーションとして実行できるようにします。             |
| <code>vdbe [global]</code>   | 仮想 DBE パラメータを設定するモードを開始します。  |
| <code>vrf vrf_name</code>  | 特定のバーチャル プライベート ネットワーク (VPN) に H.323 隣接を関連付けます。                            |