



## VRF-Aware トンネル

仮想ルーティングおよびフォワーディング（VRF） Aware トンネルは、信頼できないコア ネットワークまたは別のインフラストラクチャ（IPv4 または IPv6）を備えたコア ネットワークで区切られたカスタマー ネットワークに接続するために使用されます。

- [機能情報の確認, 1 ページ](#)
- [VRF-Aware トンネルの前提条件, 2 ページ](#)
- [VRF-Aware トンネルに関する情報, 2 ページ](#)
- [VRF-Aware IPv6 トンネルの設定方法, 3 ページ](#)
- [VRF-Aware トンネルの設定例, 13 ページ](#)
- [その他の関連資料, 20 ページ](#)
- [VRF-Aware トンネルの機能情報, 21 ページ](#)
- [VRF-Aware トンネルの前提条件, 22 ページ](#)

## 機能情報の確認

ご使用のソフトウェアリリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の注意事項と機能情報については、プラットフォームおよびソフトウェア リリースの[バグ検索ツール](#)とリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

プラットフォームのサポートおよびシスコソフトウェアイメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、[www.cisco.com/go/cfn](http://www.cisco.com/go/cfn) に移動します。Cisco.com のアカウントは必要ありません。

## VRF-Aware トンネルの前提条件

- カスタマーエッジネットワークを設定する必要があります。「[トンネリング用のカスタマーエッジネットワークの設定](#)」の項を参照してください。
- カスタマーを設定し、VRF を転送する必要があります。「[VRF インスタンスの定義](#)」の項を参照してください。

## VRF-Aware トンネルに関する情報

### トンネルの IP 送信元および宛先の VRF メンバーシップ

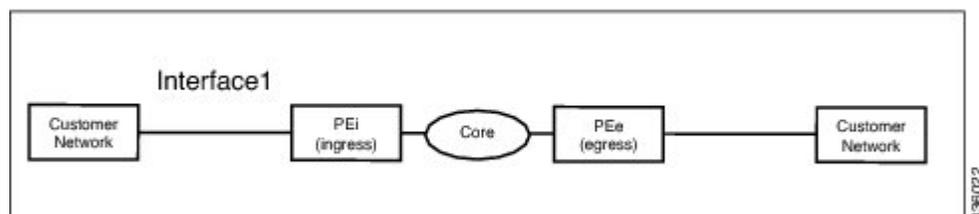
任意の VPN ルーティング/転送 (VRF) テーブルに属するようにトンネルの送信元と宛先を設定できます。VRF テーブルには、各 VPN のルーティングデータが保管されます。VRF テーブルでは、ネットワーク アクセス サーバ (NAS) に接続されているカスタマーサイトの VPN メンバーシップを定義します。各 VRF テーブルは、IP ルーティングテーブル、派生したシスコエクスプレス フォワーディング テーブル、およびルーティングテーブルに含まれる情報を制御するガイドラインおよびルーティング プロトコル パラメータから構成されます。

任意の VRF またはグローバルテーブルに属するようにトンネルの送信元と宛先を設定できます。トンネルは、トンネルの宛先へのルートが定義されていない場合は無効になります。

## VRF-Aware トンネル

仮想ルーティングおよびフォワーディング (VRF) Aware トンネルは、信頼できない IPv4 コア ネットワークまたは IPv6 コア ネットワークで区切られたカスタマー ネットワークに接続するために使用されます。

図 1: VRF-Aware トンネル



上記のトポロジでは、トンネルがコア ネットワークに設定されます。プロバイダーエッジ (PE) デバイス PEi は、インターフェイス 1 に到着したパケットのトンネルヘッドです。PE デバイス PEe は、インターフェイス 1 に到着したパケットのトンネルテールです。

インターフェイス 1 に設定された VRF はカスタマー VRF です。インターフェイス 1 を介して到着したパケットは、この VRF を使用してルーティングされます。トンネルから出たパケットはこの VRF に転送されます。カスタマー VRF によるルーティングは、内部 IP パケットルーティングと呼ばれます。

`tunnel vrf` コマンドを使用して設定された VRF はトランスポート VRF です。トランスポート VRF は、カプセル化されたペイロードに適用され、トンネルエンドポイントを調べるために使用される VRF です。この VRF は、トンネルがパケットを送信する際に経由する物理インターフェイスに関連付けられている VRF と同じです。トランスポート VRF によるルーティングは、外部 IP パケットルーティングと呼ばれます。

トンネルエンドポイントは、グローバルルーティングテーブルからのアドレスか、設定済みのトランスポート VRF テーブルからのアドレスとして設定できます。

## IPv6 トンネルを介した VRF-Aware IPv6

仮想ルーティングおよび転送 (VRF) Aware IPv6 トンネルを信頼できない IPv6 インフラストラクチャ内に作成することにより、このインフラストラクチャ上に IPv6 パケットを転送することができます。これらのトンネルは、VRF テーブルまたはグローバルルーティングテーブルにエンドポイントを持つことができます。使用されるトンネルのモードは、`tunnel mode gre ipv6` と `tunnel mode ipv6` です。

## IPv6 トンネルを介した VRF-Aware IPv4

仮想ルーティングおよび転送 (VRF) Aware IPv4 トンネルを信頼できない IPv6 インフラストラクチャ内に作成することにより、このインフラストラクチャ上に IPv4 パケットを転送することができます。これらのトンネルは、VRF テーブルまたはグローバルルーティングテーブルにエンドポイントを持つことができます。使用されるトンネルのモードは、`tunnel mode gre ipv6` と `tunnel mode ipv6` です。

## IPv4 トンネルを介した VRF-Aware IPv6

仮想ルーティングおよび転送 (VRF) Aware IPv6 トンネルを信頼できない IPv4 インフラストラクチャ内に作成することにより、このインフラストラクチャ上に IPv6 パケットを転送することができます。これらのトンネルは、VRF テーブルまたはグローバルルーティングテーブルにエンドポイントを持つことができます。使用されるトンネルのモードは、`tunnel mode gre ipv4` (デフォルトのモード) と `tunnel mode ipv4` です。

# VRF-Aware IPv6 トンネルの設定方法

VRF-Aware トンネルを設定するには、次の手順を実行する必要があります。

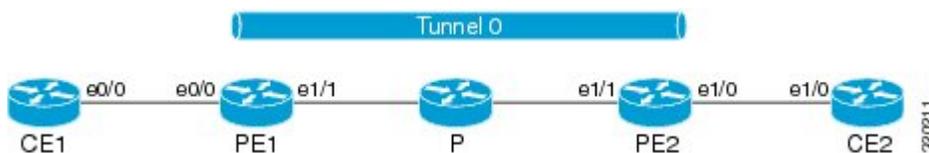
- 1 **カスタマー VRF とトランスポート VRF を定義します**：トンネルが VRF-Aware の場合はカスタマー VRF を定義します。トンネルエンドポイントを VRF に設定する必要がある場合は、トランスポート VRF を定義します。「[VRF インスタンスの定義](#)」の項を参照してください。

- 2 ネットワークをセットアップします：関連するインターフェイスを設定し、関連するルートを設定します。有効なルートが PE デバイス間とカスタマーのネットワークの間にあることを確認します。
- 3 PE デバイス間にトンネルを設定します：「[VRF-Aware トンネルの設定](#)」の項を参照してください。
  - 1 トンネルアドレスを設定します
  - 2 トンネル送信元を設定します：これは、PE デバイス上のインターフェイスです。
  - 3 トンネルの宛先を設定します：これは、他の PE デバイスのトンネルの送信元です。トンネルの正しい設定には、トンネルの宛先が ping コマンドで PE デバイスから到達可能でなければなりません（有効なルートが、トンネルの宛先に存在する必要があります）。
  - 4 トンネルモードを設定します
- 4 カスタマーエッジネットワークを設定します。「[トンネリング用のカスタマーエッジネットワークの設定](#)」の項を参照してください。
- 5 トンネルを使用してスタティックルートを設定します：設定済みのトンネルを使用してリモート CE ネットワークに PE デバイス上のルートを設定します。

## VRF-Aware トンネルの設定

このタスクでは、次のイメージに示すように、PE1 と PE2 間にトンネルを設定します。両方の PE デバイス、PE1 と PE2 について設定作業を繰り返す必要があります。

図 2: VRF-Aware トンネルの設定



## 手順の概要

1. **interface** *type number*
2. **vrf forwarding** *transport-vrf-name*
3.
  - **ip address** *ip-address mask* または
  - **ipv6 address** *ipv6-address/prefix-length*
4. **exit**
5. プロバイダー エッジ デバイス間にスタティック ルートを設定します。
6. **interface tunnel** *number*
7. **vrf forwarding** *customer-vrf-name*
8.
  - **ip address** *ip-address mask* または
  - **ipv6 address** *ipv6-address/prefix-length*
9. **tunnel source** *interface-type interface-number*
10. **tunnel destination** [*ip-address* | *ipv6-address*]
11. **tunnel vrf** *transport-vrf-name*
12. **tunnel mode** {*aurp* | *cayman* | *dvmrp* | *eon* | *gre* | *gre multipoint* | *gre ipv6* | *ipip* [*decapsulate-any*] | *ipsec ipv4* | *iptalk* | *ipv6* | *ipsec ipv6* | *mpls* | *nos* | *rbscp*}
13. **exit**
14.
  - **ip route** [*vrf vrf-name*] *prefix mask interface-type interface-number* [*next-hop-ip-address*] または
  - **ipv6 route** [*vrf vrf-name*] *destination-ipv6-prefix interface-type interface-number* [*next-hop-ipv6-address*]
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>interface</b> <i>type number</i>  例： Device(config)# interface ethernet 1/1	トンネルの送信元として使用するインターフェイスを設定します。
ステップ 2	<b>vrf forwarding</b> <i>transport-vrf-name</i>  例： Device(config-if)# vrf forwarding red	(任意) トンネルとトランスポート VRF を関連付けます。 (注) この手順は、トンネルエンドポイントがグローバルルーティングテーブルに設定されている場合には不要です。

	コマンドまたはアクション	目的
ステップ 3	<ul style="list-style-type: none"> <li>• <b>ip address</b> <i>ip-address mask</i> または</li> <li>• <b>ipv6 address</b> <i>ipv6-address/prefix-length</i></li> </ul> <p>例 :</p> <pre>Device(config-if)# ip address 10.22.22.22 255.255.255.255</pre> <p>または</p> <pre>Device(config-if)# ipv6 address 2001:DB8:3::1/64</pre>	<p>トンネルの送信元インターフェイスの IP アドレスを設定します。</p> <ul style="list-style-type: none"> <li>• PE1 について、この手順で設定されたアドレスは、トンネル エンドポイントまたはトンネルの宛先として使用されます。一方、PE2 にトンネルを設定する場合はこの逆となります。</li> <li>• このアドレスは、グローバルルーティングテーブルまたは VRF に設定されている場合があります。</li> </ul>
ステップ 4	<p><b>exit</b></p> <p>例 :</p> <pre>Device(config-if)# exit</pre>	<p>インターフェイスコンフィギュレーションモードを終了し、グローバル コンフィギュレーションモードに戻ります。</p>
ステップ 5	<p>プロバイダー エッジデバイス間にスタティック ルートを設定します。</p>	<p>プロバイダー エッジデバイスは、<b>ping</b> コマンドまたは <b>ping vrf</b> コマンドを使用して到達可能です。</p>
ステップ 6	<p><b>interface tunnel number</b></p> <p>例 :</p> <pre>Device(config)# interface tunnel 0</pre>	<p>トンネル インターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。 PE2 に同じトンネルを設定する必要があります。</p>
ステップ 7	<p><b>vrf forwarding customer-vrf-name</b></p> <p>例 :</p> <pre>Device(config-if)# vrf forwarding green</pre>	<p>(任意) トンネルとカスタマー VRF を関連付けます。</p> <ul style="list-style-type: none"> <li>• トンネルから出たパケットはこの VRF (内部 IP パケット) に転送されます。</li> </ul> <p>(注) このステップは、VRF-Aware トンネルでのみ必要です。</p>
ステップ 8	<ul style="list-style-type: none"> <li>• <b>ip address</b> <i>ip-address mask</i> または</li> <li>• <b>ipv6 address</b> <i>ipv6-address/prefix-length</i></li> </ul> <p>例 :</p> <pre>Device(config-if)# ip address 10.4.1.1 255.255.255.0</pre> <p>または</p> <pre>Device(config-if)# ipv6 address 2001:DB8:3::1/64</pre>	<p>トンネルの IPv4 アドレスまたは IPv6 アドレスを設定します。</p> <ul style="list-style-type: none"> <li>• このアドレスは、ネクストホップアドレスとしてスタティック ルートの設定時に使用されます。 PE1 と PE2 に同じネットワーク内のアドレスがあることを確認します。</li> </ul>

	コマンドまたはアクション	目的
ステップ 9	<b>tunnel source</b> <i>interface-type interface-number</i>  例： Device(config-if)# tunnel source ethernet 1/1	トンネルインターフェイスの送信元アドレスを設定します。
ステップ 10	<b>tunnel destination</b> [ <i>ip-address   ipv6-address</i> ]  例： Device(config-if)# tunnel destination 10.44.44.44	(任意) トンネルインターフェイスの宛先を指定します。  <ul style="list-style-type: none"> <li>• PE2 デバイスのトンネル送信元アドレスは PE1 のトンネル宛先アドレスとして使用されます。また、この逆の使用も可能です。</li> <li>• IPv6 インフラストラクチャが2つの PE デバイス間にある場合は、IPv6 アドレスを使用します。IPv4 インフラストラクチャが2つの PE デバイス間にある場合は、IPv4 アドレス (IPv4 トンネルを経由する IPv6) を使用します。</li> </ul>
ステップ 11	<b>tunnel vrf</b> <i>transport-vrf-name</i>  例： Device(config-if)# tunnel vrf red	(任意) トンネルとトランスポート VRF を関連付けます。  <ul style="list-style-type: none"> <li>• この VRF は、トンネルがパケット (外部 IP パケットルーティング) を送信する際に経由する物理インターフェイスに関連付けられている VRF と同じです。</li> </ul> (注) この手順は、トンネルエンドポイントがグローバルルーティングテーブルに設定されている場合には不要です。
ステップ 12	<b>tunnel mode</b> { <i>aurp   cayman   dvmrp   eon   gre   gre multipoint   gre ipv6   ipip   ipsec ipv4   ipsec ipv6   mpls   nos   rbscp</i> }  例： Device(config-if)# tunnel mode ipv6	(任意) トンネルインターフェイスのカプセル化モードを設定します。 (注) トンネルモードが GRE IPv4 の場合は、これがデフォルトモードであるため、この手順は不要です。
ステップ 13	<b>exit</b>  例： Device(config-if)# exit	インターフェイスコンフィギュレーションモードを終了し、グローバルコンフィギュレーションモードに入ります。
ステップ 14	<ul style="list-style-type: none"> <li>• <b>ip route</b> [<i>vrf vrf-name</i>] <i>prefix mask interface-type interface-number [next-hop-ip-address]</i> または</li> </ul>	設定されたトンネルを使用してリモートカスタマー ネットワークへのスタティック ルートを確立します。  <ul style="list-style-type: none"> <li>• ネクストホップとしてトンネルアドレスを使用します。</li> </ul>

	コマンドまたはアクション	目的
	<ul style="list-style-type: none"> <li>• <b>ipv6 route</b> [<b>vrf vrf-name</b>] <i>destination-ipv6-prefix interface-type interface-number</i> [<i>next-hop-ipv6-address</i>]</li> </ul> <p>例 :</p> <pre>Device(config)# ip route 10.44.44.0 255.255.255.0 10.22.22.23 Device(config)# ip route vrf red 10.44.44.0 255.255.255.0 10.22.22.23</pre> <p>または</p> <pre>Device(config)# ipv6 route 2001:DB8:2:2::/64 2001:DB8:2:1::2 Device(config)# ipv6 route vrf green 2001:DB8:2:2::/64 2001:DB8:2:1::2</pre>	<ul style="list-style-type: none"> <li>• PE1 では、ネットワーク PE2-CE2 へのスタティック ルートを設定します。PE2では、ネットワーク PE1-CE1 へのスタティック ルートを設定します。</li> </ul>
ステップ 15	<p><b>end</b></p> <p>例 :</p> <pre>Device(config)# end</pre>	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

### 次の作業

IPv6 トンネルを確認します。「[VRF-Aware トンネルの確認](#)」を参照してください。

## VRF インスタンスの定義

仮想ルーティングおよび転送 (VRF) Aware デバイスを作成し、VRF-Aware トンネルを設定するには、このタスクを実行します。

### 手順の概要

1. **vrf definition** *vrf-name*
2. **rd** *route-distinguisher*
3. **route-target export** *route-target-ext-community*
4. **route-target import** *route-target-ext-community*
5. **address-family** {*ipv4* | *ipv6*}
6. **exit-address-family**
7. **exit**

## 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>vrf definition</b> <i>vrf-name</i>  例： Device(config)# vrf definition green	VRF ルーティング テーブル インスタンスを定義するための IP VRF コンフィギュレーション モードを開始します。
ステップ 2	<b>rd</b> <i>route-distinguisher</i>  例： Device(config-vrf)# rd 1:1	VRF インスタンスのルート識別子 (RD) を指定します。
ステップ 3	<b>route-target export</b> <i>route-target-ext-community</i>  例： Device(config-vrf)# route-target export 1:1	ターゲット VPN 拡張コミュニティにルーティング情報をエクスポートします。
ステップ 4	<b>route-target import</b> <i>route-target-ext-community</i>  例： Device(config-vrf)# route-target import 1:1	ターゲット VPN 拡張コミュニティにルーティング情報をインポートします。
ステップ 5	<b>address-family</b> { <i>ipv4</i>   <i>ipv6</i> }  例： Device(config-vrf)# address-family ipv6	VRF アドレス ファミリ コンフィギュレーション モードを開始し、標準 IPv4 または IPv6 アドレス プレフィックスを使用するルーティング セッションを設定します。
ステップ 6	<b>exit-address-family</b>  例： Device(config-vrf-af)# exit-address-family	VRF アドレス ファミリ コンフィギュレーション モードを終了し、IP VRF コンフィギュレーション モードを開始します。
ステップ 7	<b>exit</b>  例： Device(config-vrf)# exit	IP VRF コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。

## トンネリング用のカスタマー エッジ ネットワークの設定

カスタマー エッジ (CE) ネットワークを設定するには、このタスクを実行します。この設定では、CE ネットワークは、プロバイダー エッジ (PE) デバイスに接続された CE デバイスを含む

ネットワークです。PE1 と CE1 を接続し、PE2 と CE2 を接続します。アドレスは適切に設定する必要があります。

### はじめる前に

カスタマー VRF を定義するには、「[VRF インスタンスの定義](#)」の項を参照してください。

### 手順の概要

1. **interface** *type number*
2. **vrf forwarding** *customer-vrf-name*
3.
  - **ip address** *ip-address mask* または
  - **ipv6 address** *ipv6-address/prefix-length*
4. **exit**

### 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>interface</b> <i>type number</i>  例： Device(config)# interface Ethernet 0/0	インターフェイスタイプを設定し、インターフェイスコンフィギュレーションモードを開始します。
ステップ 2	<b>vrf forwarding</b> <i>customer-vrf-name</i>  例： Device(config-if)# vrf forwarding green	(任意) VRF インスタンスまたは仮想ネットワークをトンネルと関連付けます。 (注) この手順は、インターフェイスを VRF に関連付ける必要がある場合のみ必要です。
ステップ 3	<ul style="list-style-type: none"> <li>• <b>ip address</b> <i>ip-address mask</i> または</li> <li>• <b>ipv6 address</b> <i>ipv6-address/prefix-length</i></li> </ul> 例： Device(config-if)# ip address 10.22.22.22 255.255.255.0 または Device(config-if)# ipv6 address 2001:DB8:1::1/64	インターフェイスのアドレスを設定します。  <ul style="list-style-type: none"> <li>• PE デバイスに接続された CE デバイスが同じネットワーク上にあることを確認します。</li> </ul>
ステップ 4	<b>exit</b>  例： Device(config-if)# exit	インターフェイスコンフィギュレーションモードを終了し、グローバルコンフィギュレーションモードに入ります。

## VRF-Aware トンネルの確認

仮想ルーティングおよび転送 (VRF) Aware トンネルを確認するには、次のコマンドを使用します。

### 手順の概要

1. **show tunnel interface**
2. **show ip route ip-address**
3. **show ip route vrf vrf-name ip-address**
4. **ping ipv6 ipv6-address source ipv6-address**
5. **ping vrf vrf-name ipv6-address source ipv6-address**
6. **debug ipv6 icmp**

### 手順の詳細

#### ステップ 1 show tunnel interface

このコマンドは、すべてのトンネル インターフェイスに関する詳細情報を表示します。

例：

次は、総称ルーティングカプセル化 (GRE) トンネルモードのプロバイダーエッジ (PE) からの出力例です。

```
Device# show tunnel interface

Tunnel0
  Mode:GRE/IP, Destination 10.44.44.44, Source Loopback2
  IP transport: output interface Ethernet1/0 next hop 10.0.0.2,
  Tunnel header destination 10.44.44.44
  Application ID 1: unspecified
  Linestate - current up, cached up
  Internal linestate - current up, evaluated up
```

例：

次は IPv6/IP トンネルモードの PE デバイスからの出力例です。

```
Device# show tunnel interface

Tunnel0
  Mode:IPv6/IP, Destination 44.44.44.44, Source Loopback2
  IP transport: output interface Ethernet1/0 next hop 2.0.0.2,
  Tunnel header destination 44.44.44.44
  Application ID 1: unspecified
  Linestate - current up, cached up
  Internal linestate - current up, evaluated up
```

出力が表示され、トンネルモードが確認できます。

#### ステップ 2 show ip route ip-address

このコマンドは、トンネルの宛先アドレスに詳細なルーティング情報を表示します。

例：

次は、グローバルルーティングテーブルでのトンネルエンドポイントのPEデバイスからの出力例です。

```
Device# show ip route 10.44.44.44

Routing entry for 10.44.44.44/32
Known via "ospf 1", distance 110, metric 21, type intra area
Last update from 10.0.0.2 on Ethernet1/0, 01:10:25 ago
Routing Descriptor Blocks:
* 10.0.0.2, from 10.44.44.44, 01:10:25 ago, via Ethernet1/0
  Route metric is 21, traffic share count is 1
```

次は、VRF テーブルにトンネル エンドポイントを持つ PE デバイスからの出力例です。

```
Device# show ip route 10.44.44.44

% Network not in table
```

出力が表示され、トンネルの宛先がグローバル ルーティング テーブルにあるかどうかを確認できます。

### ステップ3 show ip route vrf vrf-name ip-address

このコマンドは、宛先 IP アドレスに詳細なルーティング情報を表示します。

例：

次は、PE1 からの出力例です。

```
Device# show ip route vrf green 10.4.4.4

Routing entry for 10.4.4.4/32
Known via "static", distance 1, metric 0
Routing Descriptor Blocks:
* 10.0.0.2, via Ethernet1/0
  Route metric is 0, traffic share count is 1
```

トンネル宛先アドレス 10.4.4.4 は、グローバル ルーティング テーブルにありません。

### ステップ4 ping ipv6 ipv6-address source ipv6-address

このコマンドは、2つのデバイス間の接続の状態を表示します。

例：

次は、カスタマー エッジ (CE) デバイス CE1 において CE2 に対して発行した ping コマンドの出力例です。

```
Device# ping ipv6 2001:DB8:2::1 source 2001:DB8:1::1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:2::1, timeout is 2 seconds:
Packet sent with a source address of 2001:DB8:1::1
!!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/2/4 ms
```

### ステップ5 ping vrf vrf-name ipv6-address source ipv6-address

VRF-ping は VPN 接続をテストします。

例：

次は、CE2 に対して出された `ping vrf green ipv6 2001:DB8:2::1 source 2001:DB8:1::1` による CE1 からの出力例です。

```
Device# ping vrf green ipv6 2001:DB8:2::1 source 2001:DB8:1::1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:2::1, timeout is 2 seconds:
Packet sent with a source address of 2001:DB8:1::2%green
!!!!!
```

表示された出力が成功を示している場合、VPN が正しく設定されています。

## ステップ 6 debug ipv6 icmp

このコマンドは、IPv6 インターネット制御メッセージプロトコル (ICMP) トランザクションのデバッグメッセージを表示します。

例：

次にサンプル出力を示します。

```
Device# debug ipv6 icmp

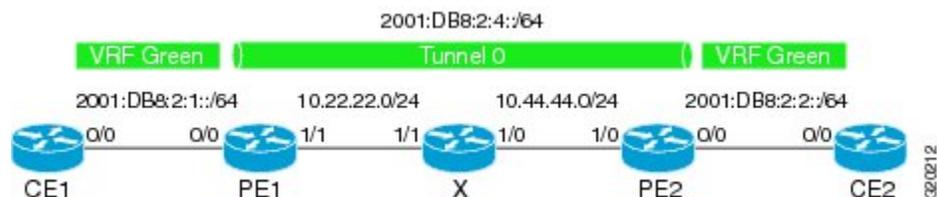
ICMP Packet debugging is on

*Apr 6 14:08:10.743: ICMPv6: Received echo request, Src=2001:DB8:1::2, Dst=2001:DB8:2::1
*Apr 6 14:08:10.743: ICMPv6: Sent echo reply, Src=2001:DB8:2::1, Dst=2001:DB8:1::2
...
```

表示された出力が成功を示している場合、VPN が正しく設定されています。

## VRF-Aware トンネルの設定例

例：VRF-Aware トンネルの設定（グローバルルーティングテーブルでのトンネルエンドポイント）



例：CE1 の設定

```
!
ipv6 unicast-routing
ipv6 cef
!
vrf definition green
rd 1:1
```

## 例 : VRF-Aware トンネルの設定 (グローバルルーティングテーブルでのトンネルエンドポイント)

```

route-target export 1:1
route-target import 1:1
address-family ipv6
exit-address-family
exit
!
interface Ethernet0/0
 vrf forwarding green
 no ip address
 ipv6 address 2001:DB8:2:1::1/64
 no shutdown
 exit
!
!
ipv6 route vrf green 2001:DB8:2:2::/64 2001:DB8:2:1::2
ipv6 route vrf green 2001:DB8:2:4::/64 2001:DB8:2:1::2
!

```

## 例 : PE1 の設定

```

ipv6 unicast-routing
ipv6 cef
!
vrf definition green
 rd 1:1
 route-target export 1:1
 route-target import 1:1
 address-family ipv6
 exit-address-family
 exit
!
interface Tunnel0
 no ip address
 vrf forwarding green
 ipv6 address 2001:DB8:2:4::1/64
 tunnel source 10.22.22.22
 tunnel destination 10.44.44.44
 exit
!
interface Ethernet0/0
 vrf forwarding green
 no ip address
 ipv6 address 2001:DB8:2:1::2/64
 no shutdown
 exit
!
interface Ethernet1/1
 no ip address
 ip address 10.22.22.22 255.255.255.0
 no shutdown
 exit
!
ip route 10.44.44.0 255.255.255.0 10.22.22.23
ipv6 route vrf green 2001:DB8:2:2::/64 Tunnel0 2001:DB8:2:4::2

```

## 例 : PE2 の設定

```

!
ipv6 unicast-routing
ipv6 cef
!
vrf definition green
 rd 1:1
 route-target export 1:1
 route-target import 1:1
 address-family ipv6
 exit-address-family
 exit

```

```

!
interface Tunnel0
 vrf forwarding green
 no ipv6 address
 ipv6 address 2001:DB8:2:4::2/64
 tunnel source 10.44.44.44
 tunnel destination 10.22.22.22
 exit
!
interface Ethernet0/0
 vrf forwarding green
 no ipv6 address
 ipv6 address 2001:DB8:2:2::1/64
 no shutdown
 exit
!
interface Ethernet1/0
 no ip address
 ip address 10.44.44.44 255.255.255.0
 no shutdown
 exit
!
ip route 10.22.22.0 255.255.255.0 10.44.44.43
!
ipv6 route vrf green 2001:DB8:2:1::/64 Tunnel0 2001:DB8:2:4::1
!

```

#### 例：CE2 の設定

```

!
ipv6 unicast-routing
ipv6 cef
!
vrf definition green
 rd 1:1
 route-target export 1:1
 route-target import 1:1
 address-family ipv6
 exit-address-family
 exit
!
interface Ethernet0/0
 vrf forwarding green
 no ipv6 address
 ipv6 address 2001:DB8:2:2::2/64
 no shutdown
 exit
!
!
ipv6 route vrf green 2001:DB8:2:1::/64 2001:DB8:2:2::1
ipv6 route vrf green 2001:DB8:2:4::/64 2001:DB8:2:2::1
!

```

#### 例：デバイス X の設定

```

!
interface Ethernet1/0
 no ip address
 ip address 10.44.44.43 255.255.255.0
 no shutdown
 exit
!
interface Ethernet1/1
 no ip address
 ip address 10.22.22.23 255.255.255.0
 no shutdown
 exit

```

例 : VRF-Aware トンネルの設定 (グローバルルーティングテーブルでのトンネルエンドポイント)

!

### 例 : トンネル設定の確認

#### CE1 から

```
Device# ping vrf green ipv6 2001:db8:2:2::2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:2:2::2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/6 ms

Device# ping vrf green ipv6 2001:db8:2:2::2 source 2001:db8:2:1::1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:2:2::2, timeout is 2 seconds:
Packet sent with a source address of 2001:DB8:2:1::1
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

#### PE1 から

```
Device# show tunnel interface

Tunnel0
  Mode:GRE/IP, Destination 10.44.44.44, Source 10.22.22.22
  IP transport: output interface Ethernet1/1 next hop 10.22.22.23
  Application ID 1: unspecified
  Linestate - current up
  Internal linestate - current up, evaluated up
  Tunnel Source Flags: Local
  Transport IPv4 Header DF bit cleared
  OCE: IP tunnel decap
  Provider: interface Tu0, prot 47
    Performs protocol check [47]
    Protocol Handler: GRE: opt 0x0
      ptype: ipv4 [ipv4 dispatcher: punt]
      ptype: ipv6 [ipv6 dispatcher: from if Tu0]
      ptype: mpls [mpls dispatcher: drop]
      ptype: otv [mpls dispatcher: drop]
      ptype: generic [mpls dispatcher: drop]
  There are 0 tunnels running over the EON IP protocol
  There are 0 tunnels running over the IPinIP protocol
  There are 0 tunnels running over the NOSIP protocol
  There are 0 tunnels running over the IPv6inIP protocol
  There are 0 tunnels running over the RBSCP/IP protocol

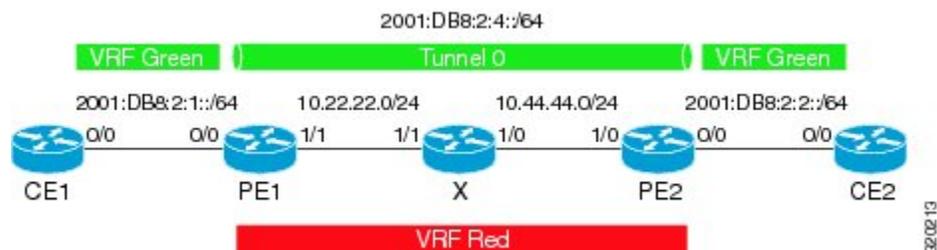
Device# show ip route 10.44.44.44

Routing entry for 10.44.44.0/24
  Known via "static", distance 1, metric 0
  Routing Descriptor Blocks:
  * 10.22.22.23
    Route metric is 0, traffic share count is 1

Device# debug ipv6 icmp

ICMP Packet debugging is on
*Jan 1 10:57:37.882: ICMPv6: Sent R-Advert, Src=FE80::A8BB:CCFF:FE00:5200, Dst=FF02::1
*Jan 1 11:00:18.634: ICMPv6: Received R-Advert, Src=FE80::A8BB:CCFF:FE00:5200, Dst=FF02::1
```

## 例：VRF-Aware トンネルの設定（VRF でのトンネルエンドポイント）



## 例：CE1 の設定

```

!
ipv6 unicast-routing
ipv6 cef
!
vrf definition green
rd 1:1
route-target export 1:1
route-target import 1:1
address-family ipv6
exit-address-family
exit
!
interface Ethernet0/0
vrf forwarding green
no ip address
ipv6 address 2001:DB8:2:1::1/64
no shutdown
exit
!
!
ipv6 route vrf green 2001:DB8:2:2::/64 2001:DB8:2:1::2
ipv6 route vrf green 2001:DB8:2:4::/64 2001:DB8:2:1::2
!

```

## 例：PE1 の設定

```

ipv6 unicast-routing
ipv6 cef
!
vrf definition green
rd 1:1
route-target export 1:1
route-target import 1:1
address-family ipv6
exit-address-family
exit
!
vrf definition red
rd 2:2
route-target export 2:2
route-target import 2:2
address-family ipv4
exit-address-family
exit
!
interface Tunnel0
no ip address
vrf forwarding green
ipv6 address 2001:DB8:2:4::1/64
tunnel source 10.22.22.22

```

## 例 : VRF-Aware トンネルの設定 (VRF でのトンネルエンドポイント)

```

tunnel destination 10.44.44.44
tunnel vrf red
exit
!
interface Ethernet0/0
vrf forwarding green
no ip address
ipv6 address 2001:DB8:2:1::2/64
no shutdown
exit
!
interface Ethernet1/1
vrf forwarding red
no ip address
ip address 10.22.22.22 255.255.255.0
no shutdown
exit
!
ip route vrf red 10.44.44.0 255.255.255.0 10.22.22.23
ipv6 route vrf green 2001:DB8:2:2::/64 Tunnel0 2001:DB8:2:4::2

```

## 例 : PE2 の設定

```

!
ipv6 unicast-routing
ipv6 cef
!
vrf definition green
rd 1:1
route-target export 1:1
route-target import 1:1
address-family ipv6
exit-address-family
exit
!
vrf definition red
rd 2:2
route-target export 2:2
route-target import 2:2
address-family ipv4
exit-address-family
exit
!
interface Tunnel0
vrf forwarding green
no ipv6 address
ipv6 address 2001:DB8:2:4::2/64
tunnel source 10.44.44.44
tunnel destination 10.22.22.22
tunnel vrf red
exit
!
interface Ethernet0/0
vrf forwarding green
no ipv6 address
ipv6 address 2001:DB8:2:2::1/64
no shutdown
exit
!
interface Ethernet1/0
vrf forwarding red
no ip address
ip address 10.44.44.44 255.255.255.0
no shutdown
exit
!
ip route vrf red 10.22.22.0 255.255.255.0 10.44.44.43
!
!
ipv6 route vrf green 2001:DB8:2:1::/64 Tunnel0 2001:DB8:2:4::1

```

!

**例：CE2 の設定**

```

!
ipv6 unicast-routing
ipv6 cef
!
vrf definition green
rd 1:1
route-target export 1:1
route-target import 1:1
address-family ipv6
exit-address-family
exit
!
interface Ethernet0/0
 vrf forwarding green
 no ipv6 address
 ipv6 address 2001:DB8:2:2::2/64
 no shutdown
 exit
!
!
ipv6 route vrf green 2001:DB8:2:1::/64 2001:DB8:2:2::1
ipv6 route vrf green 2001:DB8:2:4::/64 2001:DB8:2:2::1
!

```

**例：デバイス X の設定**

```

!
interface Ethernet1/0
 vrf forwarding red
 no ip address
 ip address 10.44.44.43 255.255.255.0
 no shutdown
 exit
!
interface Ethernet1/1
 vrf forwarding red
 no ip address
 ip address 10.22.22.23 255.255.255.0
 no shutdown
 exit
!

```

**例：トンネル設定の確認****CE1 から**

```

Device# ping vrf green ipv6 2001:db8:2:2::2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:2:2::2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/6 ms

Device# ping vrf green ipv6 2001:db8:2:2::2 source 2001:db8:2:1::1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:2:2::2, timeout is 2 seconds:
Packet sent with a source address of 2001:DB8:2:1::1
!!!!

```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

## PE1 から

```
Device# show tunnel interface
```

```
Tunnel0
  Mode:GRE/IP, Destination 10.44.44.44, Source 10.22.22.22
  IP transport: output interface Ethernet1/1 next hop 10.22.22.23
  Application ID 1: unspecified
  Linestate - current up
  Internal linestate - current up, evaluated up
  Tunnel Source Flags: Local
  Transport IPv4 Header DF bit cleared
  OCE: IP tunnel decap
  Provider: interface Tu0, prot 47
  Performs protocol check [47]
  Protocol Handler: GRE: opt 0x0
    ptype: ipv4 [ipv4 dispatcher: punt]
    ptype: ipv6 [ipv6 dispatcher: from if Tu0]
    ptype: mpls [mpls dispatcher: drop]
    ptype: otv [mpls dispatcher: drop]
    ptype: generic [mpls dispatcher: drop]
  There are 0 tunnels running over the EON IP protocol
  There are 0 tunnels running over the IPinIP protocol
  There are 0 tunnels running over the NOSIP protocol
  There are 0 tunnels running over the IPv6inIP protocol
  There are 0 tunnels running over the RBSCP/IP protocol
```

```
Device# show ip route 10.44.44.44
```

```
% Network not in table
```

```
Device# show ip route vrf red 10.44.44.44
```

```
Routing Table: red
Routing entry for 10.44.44.0/24
  Known via "static", distance 1, metric 0
  Routing Descriptor Blocks:
    * 10.22.22.23
      Route metric is 0, traffic share count is 1
```

```
Device# debug ipv6 icmp
```

```
ICMP Packet debugging is on
*Jan 1 10:57:37.882: ICMPv6: Sent R-Advert, Src=FE80::A8BB:CCFF:FE00:5200, Dst=FF02::1
*Jan 1 11:00:18.634: ICMPv6: Received R-Advert, Src=FE80::A8BB:CCFF:FE00:5200, Dst=FF02::1
```

## その他の関連資料

### 関連資料

関連項目	マニュアル タイトル
IPv6 アドレッシングと接続	『Cisco IOS IPv6 Configuration Guide』
Cisco IOS コマンド	『Cisco IOS Master Commands List, All Releases』

関連項目	マニュアル タイトル
IPv6 コマンド	『Cisco IOS IPv6 Command Reference』
Cisco IOS IPv6 機能	『Cisco IOS IPv6 Feature Mapping』

標準/RFC	タイトル
IPv6 に関する RFC	『IPv6 RFCs』

### 標準および RFC

#### シスコのテクニカル サポート

説明	リンク
シスコのサポートおよびドキュメンテーション Web サイトでは、ダウンロード可能なマニュアル、ソフトウェア、ツールなどのオンラインリソースを提供しています。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	<a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a>

## VRF-Aware トンネルの機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコソフトウェアイメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、[www.cisco.com/go/cfn](http://www.cisco.com/go/cfn) に移動します。Cisco.com のアカウントは必要ありません。

表 1: VRF-Aware トンネルの機能情報

機能名	リリース	機能情報
VRF-Aware トンネル	Cisco IOS XE Release 3.8S	<p>仮想ルーティングおよびフォワーディング (VRF) Aware トンネルは、信頼できないコアネットワークまたは別のインフラストラクチャ (IPv4 または IPv6) を備えたコアネットワークで区切られたカスタマーネットワークに接続するために使用されます。</p> <p><b>tunnel vrf</b> コマンドは、IPv6 トランスポートをサポートするように変更されました。</p>

## VRF-Aware トンネルの前提条件

- カスタマーエッジネットワークを設定する必要があります。「[トンネリング用のカスタマーエッジネットワークの設定](#)」の項を参照してください。
- カスタマーを設定し、VRF を転送する必要があります。「[VRF インスタンスの定義](#)」の項を参照してください。