Multi-VRF CE の設定

Multi-VRF CE に関する情報

バーチャル プライベート ネットワーク（VPN）は、ISP バックボーン ネットワーク上でお客様にセキュアな帯域幅共有を提供します。VPN は、共通ルーティング テーブルを共有するサイトの集合です。カスタマー サイトは、1 つまたは複数のインターフェイスでサービス プロバイダ ネットワークに接続され、サービス プロバイダは、VRF テーブルと呼ばれる VPN ルーティング テーブルと各インターフェイスを関連付けます。

スイッチが Network Advantage ライセンスで稼働している場合、スイッチはカスタマー エッジ（CE）デバイスの Multiple VPN Routing/Forwarding（Multi-VRF）インスタンスをサポートします（Multi-VRF CE）。サービス プロバイダは、Multi-VRF CE により、重複する IP アドレスで複数の VPN をサポートできます。

（注） スイッチでは、VPN のサポートのためにマルチプロトコル ラベル スイッチング（MPLS）が使用されません。

Multi-VRF CE の概要

Multi-VRF CE は、サービス プロバイダが複数の VPN をサポートし、VPN 間で IP アドレスを重複して使用できるようにする機能です。Multi-VRF CE は入力インターフェイスを使用して、さまざまな VPN のルートを区別し、1 つまたは複数のレイヤ 3 インターフェイスと各 VRF を関連付けて仮想パケット転送テーブルを形成します。VRF 内のインターフェイスは、イーサネット ポートのように物理的なもの、または VLAN SVI のように論理的なものにもできますが、複数の VRF に属すことはできません。

（注） Multi-VRF CE インターフェイスは、レイヤ 3 インターフェイスである必要があります。

Multi-VRF CE には、次のデバイスが含まれます。

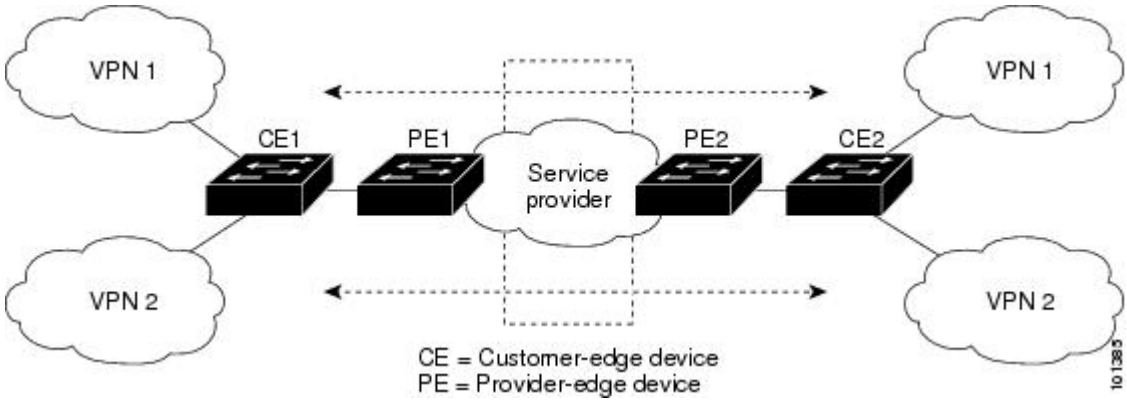
- お客様は、CEデバイスにより、1つまたは複数のプロバイダエッジ（PE）ルータへのデータ リンクを介してサービス プロバイダ ネットワークにアクセスできます。CE デバイスは、サイトのローカル ルートをルータにアドバタイズし、リモート VPN ルートをそこから学習します。スイッチを CE に設定することができます。
- PEルータは、スタティックルーティング、またはBGP、RIPv2、OSPF、EIGRPなどのルーティングプロトコルを使用して、CEデバイスとルーティング情報を交換します。PEは、直接接続している VPN に対する VPN ルートのみを保守する必要があります。そのため、すべてのサービス プロバイダ VPN ルートを PE が保守する必要はありません。各 PE ルータは、直接接続しているサイトごとに VRF を維持します。すべてのサイトが同じ VPN に存在する場合は、PE ルータの複数のインターフェイスを 1 つの VRF に関連付けることができます。各 VPN は、指定された VRF にマッピングされます。PE ルータは、ローカル VPN ルートを CE から学習したあとで、IBGP を使用して別の PE ルータと VPN ルーティング情報を交換します。
- CEデバイスに接続していないサービス プロバイダ ネットワークのルータは、プロバイダ ルータやコア ルータになります。

Multi-VRF CE では、複数のお客様が 1 つの CE を共有でき、CE と PE の間で 1 つの物理リンクだけが使用されます。共有 CE は、お客様ごとに別々の VRF テーブルを維持し、独自のルーティング テーブルに基づいて、お客様ごとにパケットをスイッチングまたはルーティングします。Multi-VRF CE は、制限付きの PE 機能を CE デバイスに拡張して、別々の VRF テーブルを維持し、VPN のプライバシーおよびセキュリティをブランチ オフィスに拡張します。

ネットワーク トポロジ

次の図に、スイッチを複数の仮想 CE として使用した構成例を示します。このシナリオは、中小企業など、VPNサービスの帯域幅要件の低いお客様に適しています。この場合、スイッチにはマルチ VRF CE のサポートが必要です。Multi-VRF CE はレイヤ 3 機能なので、VRF のそれぞれのインターフェイスはレイヤ 3 インターフェイスである必要があります。

図 *1* : 複数の仮想 *CE* として機能するスイッチ



CE スイッチは、レイヤ3インターフェイスを VRF に追加するコマンドを受信すると、Multi-VRF CE 関連のデータ構造で VLAN ID と Policy Label（PL）の間に適切なマッピングを設定し、VLAN ID と PL を VLAN データベースに追加します。

Multi-VRF CE を設定すると、レイヤ 3 フォワーディング テーブルは、次の 2 つのセクションに概念的に分割されます。

- Multi-VRF CE ルーティング セクションには、さまざまな VPN からのルートが含まれます。
- グローバル ルーティング セクションには、インターネットなど、VPN 以外のネットワークへのルートが含まれます。

さまざまな VRF の VLAN ID はさまざまな PL にマッピングされ、処理中に VRF を区別するために使用されます。レイヤ 3 設定機能では、学習した新しい VPN ルートごとに、入力ポートの VLAN ID を使用して PL を取得し、Multi-VRF CE ルーティング セクションに PL および新しいルートを挿入します。ルーテッド ポートからパケットを受信した場合は、ポート内部 VLAN ID 番号が使用されます。SVI からパケットを受信した場合は、VLAN 番号が使用されます。

パケット転送処理

Multi-VRF CE 対応ネットワークのパケット転送処理は次のとおりです。

- スイッチは、VPN からパケットを受信すると、入力 PL 番号に基づいてルーティング テーブルを検索します。ルートが見つかると、スイッチはパケットを PE に転送します。
- 入力 PE は、CE からパケットを受信すると、VRF 検索を実行します。ルートが見つかると、ルータは対応する MPLS ラベルをパケットに追加し、MPLS ネットワークに送信します。
- 出力 PE は、ネットワークからパケットを受信すると、ラベルを除去してそのラベルを使用し、正しい VPN ルーティング テーブルを識別します。次に、通常のルート検索を実行します。ルートが見つかると、パケットを正しい隣接デバイスに転送します。

- CE は、出力 PE からパケットを受信すると、入力 PL を使用して正しい VPN ルーティング テーブルを検索します。ルートが見つかると、パケットを VPN 内で転送します。

ネットワーク コンポーネント

VRF を設定するには、VRF テーブルを作成し、VRF に関連するレイヤ 3 インターフェイスを指定します。次に、VPN、および CE と PE 間でルーティング プロトコルを設定します。プロバイダのバックボーンで VPN ルーティング情報を配信する場合は、BGP が優先ルーティング プロトコルです。Multi-VRF CE ネットワークには、次の 3 つの主要コンポーネントがあります。

- VPN ルート ターゲット コミュニティ：VPN コミュニティのその他すべてのメンバのリスト。VPN コミュニティ メンバーごとに VPN ルート ターゲットを設定する必要があります。
- VPN コミュニティ PE ルータのマルチプロトコル BGP ピアリング：VPN コミュニティのすべてのメンバーに VRF 到達可能性情報を伝播します。VPN コミュニティのすべての PE ルータで BGP ピアリングを設定する必要があります。
- VPN 転送：VPN サービス プロバイダ ネットワークを介し、全 VPN コミュニティ メンバー間で、全トラフィックを伝送します。

VRF 認識サービス

IP サービスはグローバル インターフェイスに設定可能で、グローバル ルーティング インスタンスで稼働します。IP サービスは複数のルーティング インスタンス上で稼働するように拡張されます。これが、VRF 認識です。システム内の任意の設定済み VRF であればいずれも、VRF 認識サービス用に指定できます。

VRF 認識サービスは、プラットフォームに依存しないモジュールに実装されます。VRF とは、Cisco IOS 内の複数のルーティング インスタンスを意味します。各プラットフォームには、サポートする VRF 数に関して独自の制限があります。

VRF 認識サービスには、次の特性があります。

- ユーザーは、ユーザー指定の VRF 内のホストに ping を実行できます。
- ARP エントリは、個別の VRF で学習されます。ユーザーは、特定の VRF の ARP エントリを表示できます。

Multi-VRF CE の設定方法

Multi-VRF CE のデフォルト設定

表 *1 : VRF* のデフォルト設定

機能	デフォルト設定
VRF	ディセーブル。VRF は定義されていません。
マップ	インポートマップ、エクスポートマップ、ルートマップは定義されていません。
VRF 最大ルート数	ファスト イーサネット スイッチ：8000 ギガビット イーサネット スイッチ：12000
転送テーブル	インターフェイスのデフォルトは、グローバル ルーティング テーブルです。

Multi-VRF CE の設定時の注意事項

（注）

Multi-VRF CE を使用するには、スイッチで Network Advantage ライセンスをイネーブルにする必要があります。

- Multi-VRF CE を含むスイッチは複数のお客様によって共有され、各お客様には独自のルーティング テーブルがあります。
- お客様は別々の VRF テーブルを使用するので、同じ IP アドレスを再利用できます。別々の VPN では IP アドレスの重複が許可されます。
- Multi-VRF CE では、複数のお客様が、PE と CE の間で同じ物理リンクを共有できます。複数の VLAN を持つトランク ポートでは、パケットがお客様間で分離されます。それぞれのお客様には独自の VLAN があります。
- Multi-VRF CE ではサポートされない MPLS-VRF 機能があります。ラベル交換、LDP 隣接関係、ラベル付きパケットはサポートされません。
- PE ルータの場合、Multi-VRF CE の使用と複数の CE の使用に違いはありません。図 41-6 では、複数の仮想レイヤ 3 インターフェイスが Multi-VRF CE デバイスに接続されています。
- スイッチでは、物理ポートか VLAN SVI、またはその両方の組み合わせを使用して、VRF を設定できます。SVI は、アクセス ポートまたはトランク ポートで接続できます。
- お客様は、別のお客様と重複しないかぎり、複数の VLAN を使用できます。お客様の VLAN は、スイッチに保存されている適切なルーティング テーブルの識別に使用される特定のルーティング テーブル ID にマッピングされます。
- スイッチは、1 つのグローバルネットワークおよび最大 256 の VRF をサポートします。
- CE と PE の間では、ほとんどのルーティング プロトコル（BGP、OSPF、RIP、およびスタティック ルーティング）を使用できます。ただし、次の理由から External BGP（EBGP）を使用することを推奨します。
  - BGP では、複数の CE とのやり取りに複数のアルゴリズムを必要としません。
  - BGP は、さまざまな管理者によって稼働するシステム間でルーティング情報を渡すように設計されています。
  - BGP では、ルートの属性を CE に簡単に渡すことができます。
- Multi-VRF CE は、パケットのスイッチング レートに影響しません。
- VPN マルチキャストはサポートされません。
- プライベート VLAN で VRF をイネーブルにできます（逆も同様です）。
- インターフェイスでポリシーベース ルーティング（PBR）がイネーブルになっている場合は、VRF をイネーブルにできません（逆も同様です）。
- インターフェイスで Web Cache Communication Protocol（WCCP）がイネーブルになっている場合は、VRF をイネーブルにできません（逆も同様です）。

VRF の設定

次の操作を行ってください。

（注） スタック スイッチで VRF 設定を変更した場合は、スタック全体をリロードすることをお勧めします。これは、CEF と VRF コントロール プレーン間の整合性を維持し、マスター スイッチ オーバーの場合に不整合により表示されるエラー メッセージを避けるために不可欠です。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `デバイス# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ip routing** 例： `デバイス(config)# ip routing`	IP ルーティングを有効にします。
ステップ 3	**ip vrf** *vrf-name* 例： `デバイス(config)# ip vrf vpn1`	VRF 名を指定し、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 4	**rd** *route-distinguisher* 例： `デバイス(config-vrf)# rd 100:2`	ルート識別子を指定して VRF テーブルを作成します。AS 番号と任意の番号（xxx:y）または IP アドレスと任意の番号（A.B.C.D:y）を入力します。
ステップ 5	**route-target** {**export** | **import** | **both**} *route-target-ext-community* 例： `デバイス(config-vrf)# route-target both 100:2`	指定された VRF のインポート、エクスポート、またはインポートおよびエクスポート ルート ターゲット コミュニティのリストを作成します。AS システム番号と任意の番号（xxx:y）または IP アドレスと任意の番号（A.B.C.D:y）を入力します。*route-target-ext-community* は、ステップ 4 で入力した *route-distinguisher* と同一にする必要があります。
ステップ 6	**import map** *route-map* 例： `デバイス(config-vrf)# import map importmap1`	（任意）VRF にルート マップを対応付けます。

	コマンドまたはアクション	目的
ステップ 7	**interface** *interface-id* 例： `デバイス(config-vrf)# interface gigabitethernet 1/0/1`	VRF に関連付けるレイヤ 3 インターフェイスを指定し、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。インターフェイスにはルーテッド ポートまたは SVI を設定できます。
ステップ 8	**ip vrf forwarding** *vrf-name* 例： `デバイス(config-if)# ip vrf forwarding vpn1`	VRF をレイヤ3インターフェイスに対応付けます。 （注） **ip vrf forwarding**が管理インターフェイスで有効になっている場合、アクセスポイントは加入しません。
ステップ 9	**end** 例： `デバイス(config)# end`	特権 EXEC モードに戻ります。
ステップ 10	**show ip vrf** [**brief** | **detail** | **interfaces**] [*vrf-name*] 例： `デバイス# show ip vrf interfaces vpn1`	設定を確認します。設定した VRF に関する情報を表示します。
ステップ 11	**copy running-config startup-config** 例： `デバイス# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

VRF 認識サービスの設定

次のサービスは、VRF 認識です。

- ARP
- ping
- 簡易ネットワーク管理プロトコル（SNMP）
- ユニキャスト RPF（uRPF）
- Syslog
- traceroute
- FTP および TFTP

ARP 用 VRF 認識サービスの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**show ip arp vrf** *vrf-name* 例： `デバイス# show ip arp vrf vpn1`	指定された VRF 内の ARP テーブルを表示します。

ping 用 VRF 認識サービスの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**ping vrf***vrf-name***ip-host** 例： `デバイス# ping vrf vpn1 ip-host`	指定された VRF 内の ARP テーブルを表示します。

SNMP 用 VRF 認識サービスの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `デバイス# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**snmp-server trap authentication vrf** 例： `デバイス(config)# snmp-server trap authentication vrf`	VRF で、パケットに対して SNMP トラップをイネーブルにします。
ステップ 3	**snmp-server engineID remote** *host* **vrf** *vpn-instance engine-id string* 例： `デバイス(config)# snmp-server engineID remote 172.16.20.3 vrf vpn1 80000009030000B064EFE100`	スイッチ上で、リモート SNMP エンジンの名前を設定します。

	コマンドまたはアクション	目的
ステップ 4	**snmp-server host** *host* **vrf** *vpn-instance* **traps** *community* 例： `デバイス(config)# snmp-server host 172.16.20.3 vrf vpn1 traps comaccess`	SNMP トラップ動作の受信側、および SNMP トラップの送信に使用される VRF テーブルを指定します。
ステップ 5	**snmp-server host** *host* **vrf** *vpn-instance* **informs** *community* 例： `デバイス(config)# snmp-server host 172.16.20.3 vrf vpn1 informs comaccess`	SNMP 通知動作の受信先を指定し、SNMP 通知の送信に使用される VRF テーブルを指定します。
ステップ 6	**snmp-server user** *user group* **remote** *host* **vrf** *vpn-instance security model* 例： `デバイス(config)# snmp-server user abcd remote 172.16.20.3 vrf vpn1 priv v2c 3des secure3des`	SNMP アクセス用に、VRF 上にあるリモート ホストの SNMP グループにユーザーを追加します。
ステップ 7	**end** 例： `デバイス(config-if)# end`	特権 EXEC モードに戻ります。

NTP 用 VRF 認識サービスの設定

NTP 用の VRF 認識サービスの設定には、NTP サーバーと、NTP サーバーに接続された NTP クライアント インターフェイスの設定が含まれます。

始める前に

NTP クライアントとサーバーの間の接続を確認します。NTP サーバーに接続されているクライアント インターフェイスで有効な IP アドレスおよびサブネットを設定します。

NTP クライアントでの NTP 用 VRF 認識サービスの設定

NTP サーバーに接続されているクライアント インターフェイスで次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例：	特権 EXEC モードを有効にします。

	コマンドまたはアクション	目的
	Device>**enable**	• プロンプトが表示されたらパスワードを入力します。
ステップ 2	**configure terminal** 例： Device#**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： Device(config)#**interface gigabitethernet 1/0/1**	VRF に関連付けるレイヤ 3 インターフェイスを指定し、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 4	**vrf forwarding** *vrf-name* 例： Device(config-if)#**vrf forwarding A**	VRF をレイヤ3インターフェイスに対応付けます。
ステップ 5	**ip address** *ip-address subnet-mask* 例： Device(config-if)#**ip address 1.1.1.1 255.255.255.0**	インターフェイスの IP アドレスを入力します。
ステップ 6	**no shutdown** 例： Device(config-if)#**no shutdown**	インターフェイスをイネーブルにします。
ステップ 7	**exit** 例： Device(config-if)**exit**	インターフェイス コンフィギュレーション モードを終了します。
ステップ 8	**ntp authentication-key** *number* **md5** *md5-number* 例： Device(config)#**ntp authentication-key 1 md5 cisco123**	認証キーを定義します。デバイスが時刻源と同期するのは、時刻源がこれらの認証キーのいずれかを持ち、 **ntp trusted-key number** コマンドによってキー番号が指定されている場合だけです。 （注） 認証キー番号と MD5 パスワードは、クライアントとサーバーの両方で同じである必要があります。
ステップ 9	**ntp authenticate** 例： Device(config)#**ntp authenticate**	NTP 認証機能をイネーブルにします。NTP 認証はデフォルトでディセーブルになっています。

	コマンドまたはアクション	目的
ステップ 10	**ntp trusted-key** *key-number* 例： `Device(config)#ntp trusted-key 1`	NTP クライアントで同期をとれるようにするために、NTP サーバーによってその NTP パケットで提供される必要がある 1 つ以上のキーを指定します。trusted key の範囲は 1 ～ 65535 です。このコマンドにより、NTP クライアントが、信頼されていない NTP サーバーと誤って同期する、ということが防止されます。
ステップ 11	**ntp server vrf** *vrf-name* 例： `Device(config)#ntp server vrf A 1.1.1.2 key 1`	指定された VRF で NTP サーバーを設定します。

NTP サーバーでの NTP 用 VRF 認識サービスの設定

NTP サーバーで次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device>enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device#configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ntp authentication-key** *number* **md5** *passowrd* 例： `Device(config)#ntp authentication-key 1 md5 cisco123`	認証キーを定義します。デバイスが時刻源と同期するのは、時刻源がこれらの認証キーのいずれかを持ち、 **ntp trusted-key number** コマンドによってキー番号が指定されている場合だけです。 （注） 認証キー番号と MD5 パスワードは、クライアントとサーバーの両方で同じである必要があります。
ステップ 4	**ntp authenticate** 例： `Device(config)#ntp authenticate`	NTP 認証機能をイネーブルにします。NTP 認証はデフォルトでディセーブルになっています。

	コマンドまたはアクション	目的
ステップ 5	**ntp trusted-key** *key-number* 例： `Device(config)#ntp trusted-key 1`	NTP クライアントで同期をとれるようにするために、NTP サーバーによってその NTP パケットで提供される必要がある 1 つ以上のキーを指定します。trusted key の範囲は 1 ～ 65535 です。このコマンドにより、NTP クライアントが、信頼されていない NTP サーバーと誤って同期する、ということが防止されます。
ステップ 6	**interface** *interface-id* 例： `Device(config)#interface gigabitethernet 1/0/3`	VRF に関連付けるレイヤ 3 インターフェイスを指定し、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 7	**vrf forwarding** *vrf-name* 例： `Device(config-if)#vrf forwarding A`	VRF をレイヤ 3 インターフェイスに対応付けます。
ステップ 8	**ip address** *ip-address subnet-mask* 例： `Device(config-if)#ip address 1.1.1.2 255.255.255.0`	インターフェイスの IP アドレスを入力します。
ステップ 9	**exit** 例： `Device(config-if)exit`	インターフェイス コンフィギュレーション モードを終了します。

uRPF 用 VRF 認識サービスの設定

uRPF は、VRF に割り当てられたインターフェイス上で設定でき、送信元検索が VRF テーブルで実行されます。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `デバイス# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface***interface-id* 例： `デバイス(config)# interface gigabitethernet 1/0/1`	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、設定するレイヤ 3 インターフェイスを指定します。

	コマンドまたはアクション	目的
ステップ 3	**no switchport** 例： `デバイス(config-if)# no switchport`	レイヤ 2 コンフィギュレーション モードからインターフェイスを削除します（物理インターフェイスの場合）。
ステップ 4	**ip vrf forwarding** *vrf-name* 例： `デバイス(config-if)# ip vrf forwarding vpn2`	インターフェイス上で VRF を設定します。
ステップ 5	**ip address** *ip-address* 例： `デバイス(config-if)# ip address 10.1.5.1`	インターフェイスの IP アドレスを入力します。
ステップ 6	**ip verify unicast reverse-path** 例： `デバイス(config-if)# ip verify unicast reverse-path`	インターフェイス上で uRPF を有効にします。
ステップ 7	**end** 例： `デバイス(config-if)# end`	特権 EXEC モードに戻ります。

VRF 認識 RADIUS の設定

VRF 認識 RADIUS を設定するには、まず RADIUS サーバー上で AAA をイネーブルにする必要があります。『Per VRF AAA Feature Guide』で説明されているとおり、スイッチで **ip vrf forwarding** *vrf-name* サーバーグループ コンフィギュレーション コマンドと **ip radius source-interface** グローバル コンフィギュレーション コマンドがサポートされます。

syslog 用 VRF 認識サービスの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `デバイス# configure terminal`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	**logging on** 例： `デバイス(config)# logging on`	ストレージ ルータ イベント メッセージのロギングを、イネーブルまたは一時的にディセーブルにします。
ステップ 3	**logging host** *ip-address* **vrf** *vrf-name* 例： `デバイス(config)# logging host 10.10.1.0 vrf vpn1`	ロギング メッセージが送信される Syslog サーバーのホスト アドレスを指定します。
ステップ 4	**logging buffered** *logging buffered size* **debugging** 例： `デバイス(config)# logging buffered critical 6000 debugging`	メッセージを内部バッファにロギングします。
ステップ 5	**logging trap debugging** 例： `デバイス(config)# logging trap debugging`	Syslog サーバーに送信されるロギング メッセージを制限します。
ステップ 6	**logging facility** *facility* 例： `デバイス(config)# logging facility user`	ロギング ファシリティにシステム ロギング メッセージを送信します。
ステップ 7	**end** 例： `デバイス(config-if)# end`	特権 EXEC モードに戻ります。

traceroute 用 VRF 認識サービスの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**traceroute vrf** *vrf-name ipaddress* 例： `デバイス(config)# traceroute vrf vpn2 10.10.1.1`	宛先アドレスを取得する VPN VRF の名前を指定します。

FTP および TFTP 用 VRF 認識サービスの設定

FTP および TFTP を VRF 認識とするには、いくつかの FTP/TFTP CLI を設定する必要があります。たとえば、インターフェイスに付加される VRF テーブルを使用する場合、E1/0 であれば、ip tftp source-interface E1/0 コマンドまたは ip ftp source-interface E1/0 コマンドを設定して、特定のルーティング テーブルを使用するように TFTP または FTP サーバーに通知する必要があります。この例では、VRF テーブルが宛先 IP アドレスを検索するのに使用されます。これらの変更には下位互換性があり、既存の動作には影響を及ぼしません。つまり、VRF がそのインターフェイスに設定されていない場合でも、送信元インターフェイス CLI を使用して、特定のインターフェイスにパケットを送信できます。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `デバイス# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ip ftp source-interface** *interface-type interface-number* 例： `デバイス(config)# ip ftp source-interface gigabitethernet 1/0/2`	FTP 接続の発信元 IP アドレスを指定します。
ステップ 3	**end** 例： `デバイス(config)#end`	特権 EXEC モードに戻ります。
ステップ 4	**configure terminal** 例： `デバイス# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 5	**ip tftp source-interface** *interface-type interface-number* 例： `デバイス(config)# ip tftp source-interface gigabitethernet 1/0/2`	TFTP 接続用の送信元 IP アドレスを指定します。
ステップ 6	**end** 例： `デバイス(config)# end`	特権 EXEC モードに戻ります。

マルチキャスト VRF の設定

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： デバイス# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ip routing** 例： デバイス(config)# ip routing	IP ルーティング モードをイネーブルにします
ステップ 3	**ip vrf** *vrf-name* 例： デバイス(config)# ip vrf vpn1	VRF 名を指定し、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 4	**rd** *route-distinguisher* 例： デバイス(config-vrf)# rd 100:2	ルート識別子を指定して VRF テーブルを作成します。AS 番号と任意の番号（xxx:y）または IP アドレスと任意の番号（A.B.C.D:y）を入力します。
ステップ 5	**route-target** {**export** | **import** | **both**} *route-target-ext-community* 例： デバイス(config-vrf)# route-target import 100:2	指定された VRF のインポート、エクスポート、またはインポートおよびエクスポート ルート ターゲット コミュニティのリストを作成します。AS システム番号と任意の番号（xxx:y）または IP アドレスと任意の番号（A.B.C.D:y）を入力します。*route-target-ext-community* は、ステップ 4 で入力した *route-distinguisher* と同一にする必要があります。
ステップ 6	**import map** *route-map* 例： デバイス(config-vrf)# import map importmap1	（任意）VRF にルート マップを対応付けます。
ステップ 7	**ip multicast-routing vrf** *vrf-name* **distributed** 例： デバイス(config-vrf)# ip multicast-routing vrf vpn1 distributed	（任意）VRF テーブルでグローバル マルチキャスト ルーティングをイネーブルにします。

	コマンドまたはアクション	目的
ステップ 8	**interface** *interface-id* 例： `デバイス(config-vrf)# interface gigabitethernet 1/0/2`	VRF に関連付けるレイヤ 3 インターフェイスを指定し、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。インターフェイスはルーテッド ポートまたは SVI に設定できます。
ステップ 9	**ip vrf forwarding** *vrf-name* 例： `デバイス(config-if)# ip vrf forwarding vpn1`	VRF をレイヤ3インターフェイスに対応付けます。
ステップ 10	**ip address** *ip-address* **mask** 例： `デバイス(config-if)# ip address 10.1.5.1 255.255.255.0`	レイヤ 3 インターフェイスの IP アドレスを設定します。
ステップ 11	**ip pim sparse-dense mode** 例： `デバイス(config-if)# ip pim sparse-dense mode`	VRF に関連付けられているレイヤ 3 インターフェイス上で、PIM をイネーブルにします。
ステップ 12	**end** 例： `デバイス(config)# end`	特権 EXEC モードに戻ります。
ステップ 13	**show ip vrf** [**brief** | **detail** | **interfaces**] [*vrf-name*] 例： `デバイス# show ip vrf detail vpn1`	設定を確認します。設定した VRF に関する情報を表示します。
ステップ 14	**copy running-config startup-config** 例： `デバイス# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

VPN ルーティング セッションの設定

VPN内のルーティングは、サポートされている任意のルーティングプロトコル（RIP、OSPF、EIGRP、BGP）、またはスタティックルーティングで設定できます。ここで説明する設定はOSPF のものですが、その他のプロトコルでも手順は同じです。

（注） VRF インスタンス内で EIGRP ルーティングプロセスが実行されるように設定するには、**autonomous-system** *autonomous-system-number* アドレス ファミリ コンフィギュレーション モード コマンドを入力して、自律システム番号を設定する必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `デバイス# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**router ospf** *process-id* **vrf** *vrf-name* 例： `デバイス(config)# router ospf 1 vrf vpn1`	OSPF ルーティングをイネーブルにして VPN 転送テーブルを指定し、ルータ コンフィギュレーション モードを開始します。
ステップ 3	**log-adjacency-changes** 例： `デバイス(config-router)# log-adjacency-changes`	（任意）隣接ステートの変更を記録します。これは、デフォルトの状態です。
ステップ 4	**redistribute bgp** *autonomous-system-number* **subnets** 例： `デバイス(config-router)# redistribute bgp 10 subnets`	BGP ネットワークから OSPF ネットワークに情報を再配布するようにスイッチを設定します。
ステップ 5	**network** *network-number* **area** *area-id* 例： `デバイス(config-router)# network 1 area 2`	OSPF が動作するネットワーク アドレスとマスク、およびそのネットワーク アドレスのエリア ID を定義します。
ステップ 6	**end** 例： `デバイス(config-router)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show ip ospf** *process-id* 例： `デバイス# show ip ospf 1`	OSPF ネットワークの設定を確認します。

	コマンドまたはアクション	目的
ステップ 8	**copy running-config startup-config** 例： `デバイス# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

BGP PE/CE ルーティング セッションの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `デバイス# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**router bgp** *autonomous-system-number* 例： `デバイス(config)# router bgp 2`	その他の BGP ルータに AS 番号を渡す BGP ルーティング プロセスを設定し、ルータ コンフィギュレーション モードを開始します。
ステップ 3	**network** *network-number* **mask** *network-mask* 例： `デバイス(config-router)# network 5 mask 255.255.255.0`	BGP を使用してアナウンスするネットワークおよびマスクを指定します。
ステップ 4	**redistribute ospf** *process-id* **match internal** 例： `デバイス(config-router)# redistribute ospf 1 match internal`	OSPF内部ルートを再配布するようにスイッチを設定します。
ステップ 5	**network** *network-number* **area** *area-id* 例： `デバイス(config-router)# network 5 area 2`	OSPF が動作するネットワーク アドレスとマスク、およびそのネットワーク アドレスのエリア ID を定義します。
ステップ 6	**address-family ipv4 vrf** *vrf-name* 例： `デバイス(config-router)# address-family ipv4 vrf vpn1`	PE/CE ルーティング セッションの BGP パラメータを定義し、VRF アドレス ファミリ 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 7	**neighbor** *address* **remote-as** *as-number* 例： `デバイス(config-router)# neighbor 10.1.1.2 remote-as 2`	PE と CE ルータの間の BGP セッションを定義します。
ステップ 8	**neighbor** *address* **activate** 例： `デバイス(config-router)# neighbor 10.2.1.1 activate`	IPv4 アドレス ファミリのアドバタイズメントをアクティブ化します。
ステップ 9	**end** 例： `デバイス(config-router)# end`	特権 EXEC モードに戻ります。
ステップ 10	**show ip bgp** [**ipv4**] [**neighbors**] 例： `デバイス# show ip bgp ipv4 neighbors`	BGP 設定を確認します。
ステップ 11	**copy running-config startup-config** 例： `デバイス# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

Multi-VRF CE のモニタリング

表 *2 : Multi-VRF CE* 情報を表示するコマンド

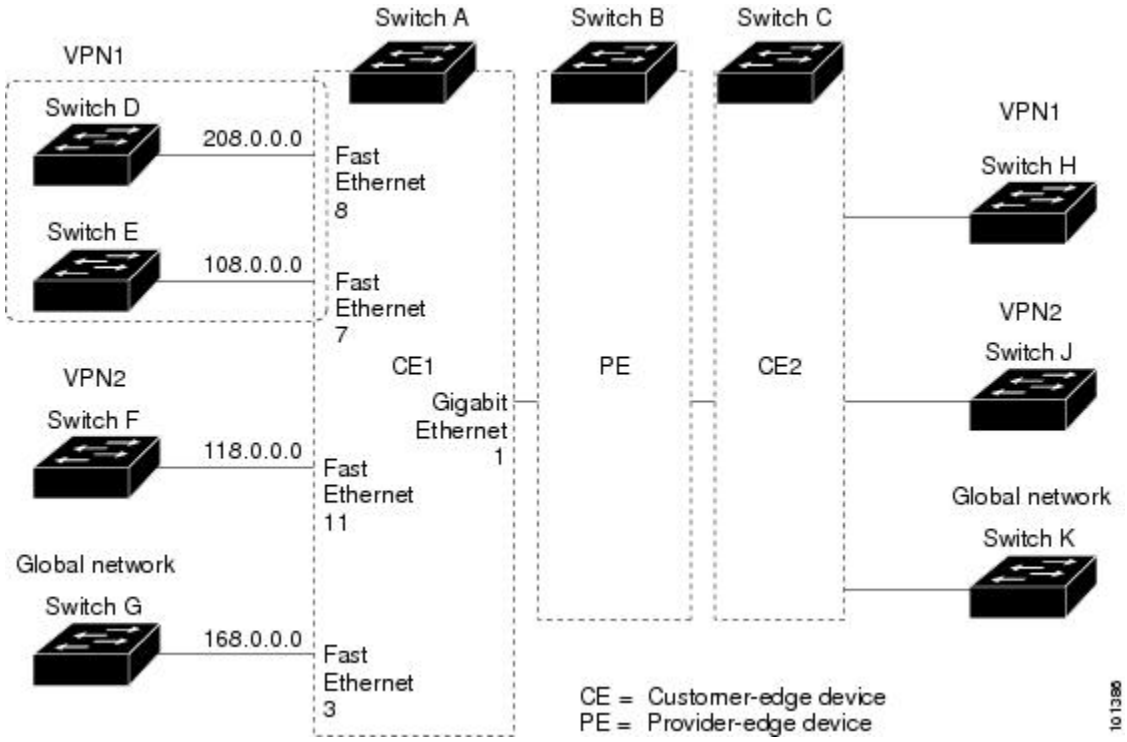
show ip protocols vrf *vrf-name*	VRF に対応付けられたルーティング情報を表示します。
show ip route vrf *vrf-name* [**connected**] [*protocol* [*as-number*]] [**list**] [**mobile**] [**odr**] [**profile**] [**static**] [**summary**] [**supernets-only**]	VRF に対応付けられた IP ルーティン情報を表示します。
show ip vrf [**brief** | **detail** | **interfaces**] [*vrf-name*]	定義された VRF インスタンスに関す示します。

Multi-VRF CE の設定例

Multi-VRF CE の設定例

VPN1、VPN2、およびグローバル ネットワークで使用されるプロトコルは OSPF です。CE/PE 接続には BGP が使用されます。図のあとに続く出力は、スイッチを CE スイッチ A として設定する例、およびカスタマー スイッチ D と F の VRF 設定を示しています。CE スイッチ C とその他のカスタマー スイッチを設定するコマンドは含まれていませんが、内容は同様です。

図 *2 : Multi-VRF CE* の設定例



スイッチ A では、ルーティングをイネーブルにして VRF を設定します。

```
デバイス# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
デバイス(config)# ip routing
デバイス(config)# ip vrf v11
デバイス(config-vrf)# rd 800:1
デバイス(config-vrf)# route-target export 800:1
デバイス(config-vrf)# route-target import 800:1
デバイス(config-vrf)# exit
デバイス(config)# ip vrf v12
デバイス(config-vrf)# rd 800:2
デバイス(config-vrf)# route-target export 800:2
```

```
デバイス(config-vrf)# route-target import 800:2
デバイス(config-vrf)# exit
```

スイッチAのループバックおよび物理インターフェイスを設定します。ギガビットイーサネット ポート 1 は PE へのトランク接続です。ギガビット イーサネット ポート 8 と 11 は VPN に接続されます。

```
デバイス(config)# interface loopback1
デバイス(config-if)# ip vrf forwarding v11
デバイス(config-if)# ip address 8.8.1.8 255.255.255.0
デバイス(config-if)# exit

デバイス(config)# interface loopback2
デバイス(config-if)# ip vrf forwarding v12
デバイス(config-if)# ip address 8.8.2.8 255.255.255.0
デバイス(config-if)# exit

デバイス(config)# interface gigabitethernet1/0/5
デバイス(config-if)# switchport trunk encapsulation dot1q
デバイス(config-if)# switchport mode trunk
デバイス(config-if)# no ip address
デバイス(config-if)# exit
デバイス(config)# interface gigabitethernet1/0/8
デバイス(config-if)# switchport access vlan 208
デバイス(config-if)# no ip address
デバイス(config-if)# exit
デバイス(config)# interface gigabitethernet1/0/11
デバイス(config-if)# switchport trunk encapsulation dot1q
デバイス(config-if)# switchport mode trunk
デバイス(config-if)# no ip address
デバイス(config-if)# exit
```

スイッチ A で使用する VLAN を設定します。VLAN 10 は、CE と PE 間の VRF 11 によって使用されます。VLAN 20 は、CE と PE 間の VRF 12 によって使用されます。VLAN 118 と 208 は、それぞれスイッチ F とスイッチ D を含む VPN に使用されます。

```
デバイス(config)# interface vlan10
デバイス(config-if)# ip vrf forwarding v11
デバイス(config-if)# ip address 38.0.0.8 255.255.255.0
デバイス(config-if)# exit
デバイス(config)# interface vlan20
デバイス(config-if)# ip vrf forwarding v12
デバイス(config-if)# ip address 83.0.0.8 255.255.255.0
デバイス(config-if)# exit
デバイス(config)# interface vlan118
デバイス(config-if)# ip vrf forwarding v12
デバイス(config-if)# ip address 118.0.0.8 255.255.255.0
デバイス(config-if)# exit
デバイス(config)# interface vlan208
デバイス(config-if)# ip vrf forwarding v11
デバイス(config-if)# ip address 208.0.0.8 255.255.255.0
デバイス(config-if)# exit
```

VPN1 と VPN2 で OSPF ルーティングを設定します。

```
デバイス(config)# router ospf 1 vrf vl1
デバイス(config-router)# redistribute bgp 800 subnets
デバイス(config-router)# network 208.0.0.0 0.0.0.255 area 0
デバイス(config-router)# exit
デバイス(config)# router ospf 2 vrf vl2
デバイス(config-router)# redistribute bgp 800 subnets
デバイス(config-router)# network 118.0.0.0 0.0.0.255 area 0
デバイス(config-router)# exit
```

CE/PE ルーティングに BGP を設定します。

```
デバイス(config)# router bgp 800
デバイス(config-router)# address-family ipv4 vrf vl2
デバイス(config-router-af)# redistribute ospf 2 match internal
デバイス(config-router-af)# neighbor 83.0.0.3 remote-as 100
デバイス(config-router-af)# neighbor 83.0.0.3 activate
デバイス(config-router-af)# network 8.8.2.0 mask 255.255.255.0
デバイス(config-router-af)# exit
デバイス(config-router)# address-family ipv4 vrf vl1
デバイス(config-router-af)# redistribute ospf 1 match internal
デバイス(config-router-af)# neighbor 38.0.0.3 remote-as 100
デバイス(config-router-af)# neighbor 38.0.0.3 activate
デバイス(config-router-af)# network 8.8.1.0 mask 255.255.255.0
デバイス(config-router-af)# end
```

スイッチ D は VPN 1 に属します。次のコマンドを使用して、スイッチ A への接続を設定します。

```
デバイス# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
デバイス(config)# ip routing
デバイス(config)# interface gigabitethernet1/0/2
デバイス(config-if)# no switchport
デバイス(config-if)# ip address 208.0.0.20 255.255.255.0
デバイス(config-if)# exit

デバイス(config)# router ospf 101
デバイス(config-router)# network 208.0.0.0 0.0.0.255 area 0
デバイス(config-router)# end
```

スイッチ F は VPN 2 に属します。次のコマンドを使用して、スイッチ A への接続を設定します。

```
デバイス# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
デバイス(config)# ip routing
デバイス(config)# interface gigabitethernet1/0/1
デバイス(config-if)# switchport trunk encapsulation dot1q
デバイス(config-if)# switchport mode trunk
デバイス(config-if)# no ip address
デバイス(config-if)# exit
```

```
デバイス(config)# interface vlan118
デバイス(config-if)# ip address 118.0.0.11 255.255.255.0
デバイス(config-if)# exit

デバイス(config)# router ospf 101
デバイス(config-router)# network 118.0.0.0 0.0.0.255 area 0
デバイス(config-router)# end
```

このコマンドをスイッチB（PEルータ）で使用すると、CEデバイス、スイッチAに対する接続だけが設定されます。

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# ip vrf v1
Router(config-vrf)# rd 100:1
Router(config-vrf)# route-target export 100:1
Router(config-vrf)# route-target import 100:1
Router(config-vrf)# exit

Router(config)# ip vrf v2
Router(config-vrf)# rd 100:2
Router(config-vrf)# route-target export 100:2
Router(config-vrf)# route-target import 100:2
Router(config-vrf)# exit
Router(config)# ip cef
Router(config)# interface Loopback1
Router(config-if)# ip vrf forwarding v1
Router(config-if)# ip address 3.3.1.3 255.255.255.0
Router(config-if)# exit

Router(config)# interface Loopback2
Router(config-if)# ip vrf forwarding v2
Router(config-if)# ip address 3.3.2.3 255.255.255.0
Router(config-if)# exit

Router(config)# interface gigabitethernet1/1/0.10
Router(config-if)# encapsulation dot1q 10
Router(config-if)# ip vrf forwarding v1
Router(config-if)# ip address 38.0.0.3 255.255.255.0
Router(config-if)# exit

Router(config)# interface gigabitethernet1/1/0.20
Router(config-if)# encapsulation dot1q 20
Router(config-if)# ip vrf forwarding v2
Router(config-if)# ip address 83.0.0.3 255.255.255.0
Router(config-if)# exit

Router(config)# router bgp 100
Router(config-router)# address-family ipv4 vrf v2
Router(config-router-af)# neighbor 83.0.0.8 remote-as 800
Router(config-router-af)# neighbor 83.0.0.8 activate
Router(config-router-af)# network 3.3.2.0 mask 255.255.255.0
Router(config-router-af)# exit
Router(config-router)# address-family ipv4 vrf vl
Router(config-router-af)# neighbor 38.0.0.8 remote-as 800
Router(config-router-af)# neighbor 38.0.0.8 activate
Router(config-router-af)# network 3.3.1.0 mask 255.255.255.0
Router(config-router-af)# end
```

マルチ VRF CE の機能情報

表 *3 :* マルチ *VRF CE* の機能情報

機能名	リリース	機能情報
マルチ VRF CE	Cisco IOS XE Everest 16.5.1a	この機能が導入されました

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。