



Easy Virtual Network コンフィギュレーション ガイド、Cisco IOS XE Release 3S

Easy Virtual Network Configuration Guide, Cisco IOS XE Release 3S

【注意】シスコ製品をご使用になる前に、安全上の注意
(www.cisco.com/jp/go/safety_warning/)をご確認ください。

本書は、米国シスコシステムズ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。

また、契約等の記述については、弊社販売パートナー、または、弊
社担当者にご確認ください。

このマニュアルに記載されている仕様および製品に関する情報は、予告なしに変更されることがあります。このマニュアルに記載されている表現、情報、および推奨事項は、すべて正確であると考えていますが、明示的であれ黙示的であれ、一切の保証の責任を負わないものとします。このマニュアルに記載されている製品の使用は、すべてユーザ側の責任になります。

対象製品のソフトウェア ライセンスおよび限定保証は、製品に添付された『Information Packet』に記載されています。添付されていない場合には、代理店にご連絡ください。

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

ここに記載されている他のいかなる保証にもよらず、各社のすべてのマニュアルおよびソフトウェアは、障害も含めて「現状のまま」として提供されます。シスコシステムズおよびこれら各社は、商品性の保証、特定目的への準拠の保証、および権利を侵害しないことに関する保証、あるいは取引過程、使用、取引慣行によって発生する保証をはじめとする、明示されたまたは黙示された一切の保証の責任を負わないものとします。

いかなる場合においても、シスコシステムズおよびその供給者は、このマニュアルの使用または使用できないことによって発生する利益の損失やデータの損傷をはじめとする、間接的、派生的、偶発的、あるいは特殊な損害について、あらゆる可能性がシスコシステムズまたはその供給者に知らされていても、それらに対する責任を一切負わないものとします。

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at www.cisco.com/go/trademarks. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1005R)

このマニュアルで使用している IP アドレスおよび電話番号は、実際のアドレスおよび電話番号を示すものではありません。マニュアル内の例、コマンド出力、ネットワーク トポロジ図、およびその他の図は、説明のみを目的として使用されています。説明の中に実際のアドレスおよび電話番号が使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

Easy Virtual Network コンフィギュレーション ガイド、Cisco IOS XE Release 3S

© 2010 Cisco Systems, Inc.

All rights reserved.

Copyright © 2010–2011, シスコシステムズ合同会社.

All rights reserved.



Easy Virtual Network の概要

Easy Virtual Network (EVN) は、複数のレイヤ 3 ネットワークのエンドツーエンドの仮想化を実現する IP ベースの仮想化テクノロジーです。単一の IP インフラストラクチャを使用して、トラフィックパスが相互に独立した状態で、個別の仮想ネットワークを提供できます。

EVN は、VRF-Lite と呼ばれる既存の IP ベースの仮想化メカニズムに基づいて構築されています。EVN はパス分離の拡張機能、簡単な設定と管理、改善された共有サービス サポートを提供します。EVN は VRF-Lite ソリューションと下位互換性があり、VRF-Lite から EVN へのシームレスなネットワーク移行を可能にします。

EVN はユニキャストルーティングの場合に IPv4、スタティック ルート、OSPFv2、および EIGRP をサポートし、IPv4 マルチキャストルーティングの場合に PIM および MSDP をサポートします。EVN は CEF と SNMP もサポートします。

機能情報の検索

ご使用のソフトウェア リリースによっては、このモジュールに記載されている機能の中に、一部サポートされていないものがあります。最新の機能情報と注意事項については、ご使用のプラットフォームとソフトウェア リリースに対応したリリース 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、および各機能がサポートされているリリースのリストについては、「[Easy Virtual Network の概要の機能情報](#)」(P.19) を参照してください。

Cisco Feature Navigator を使用すると、プラットフォーム、および Cisco ソフトウェア イメージの各サポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。Cisco.com のアカウントは必要ありません。

目次

- 「EVN の設定に関する前提条件」(P.2)
- 「EVN の制約事項」(P.2)
- 「EVN の概要」(P.2)
- 「参考資料」(P.17)
- 「[Easy Virtual Network の概要の機能情報](#)」(P.19)

EVN の設定に関する前提条件

- ネットワークへの EVN の実装には、EVN を利用して、複数の論理ネットワークまたは L3VPN に仮想化する単一の IP インフラストラクチャが必要です。EVN は異なる仮想ネットワーク上のトラフィックのパス分離を提供します。
- ネットワークに仮想化を追加する前に、機能するキャンパス設計が確立されている必要があります。
- **Virtual Routing and Forwarding (VRF)** インスタンスおよびそれらを使用して、ネットワーク全体でのトラフィックの分割を維持する方法について理解する必要があります。

EVN の制約事項

- EVN トランクは、ファスト イーサネットやギガビット イーサネットなどの 802.1q カプセル化をサポートするすべてのインターフェイスで使用できます。
- トランクには追加のプラットフォームとラインカードの制約事項があります。サポートされるプラットフォームとラインカードについては、**Cisco Feature Navigator** を確認する必要があります。
- 単一の IP インフラストラクチャを仮想化して、最大 32 のエンドツーエンドの仮想ネットワークを提供できます。
- インターフェイスに EVN トランクを設定した場合、同じインターフェイスに VRF-Lite を設定できません。
- OSPFv3 はサポートされていません。OSPFv2 はサポートされています。
- 次はサポートされていません。
 - IS-IS
 - RIP
 - BGP ではルート レプリケーションがサポートされない
 - 特定の SNMP set 操作
- 次は、EVN トランクでサポートされていません。
 - Access Control List (ACL; アクセス コントロール リスト)
 - BGP インターフェイス コマンドは継承されない
 - IPv6 (ただし、IPv6 は vnet global でサポートされる)
 - NAT
 - NetFlow
 - WCCP

EVN の概要

- 「EVN の利点」 (P.3)
- 「仮想ネットワーク タグによるパスの分離の提供」 (P.4)
- 「仮想ネットワーク タグ」 (P.6)
- 「vnet Global」 (P.7)

- 「エッジインターフェイスと EVN トランク インターフェイス」 (P.7)
- 「定義されている VRF と トランク インターフェイス上で実行する VRF 間の関係」 (P.9)
- 「VRF 認識」 (P.10)
- 「EVN でサポートされるルーティング プロトコル」 (P.10)
- 「仮想ネットワークのパケットフロー」 (P.11)
- 「EVN トランク インターフェイスでのコマンド継承」 (P.12)
- 「コマンド継承の上書き：仮想ネットワーク インターフェイス モード」 (P.13)
- 「EXEC ルーティング コンテキスト」 (P.15)
- 「EVN の VRF-Lite との互換性」 (P.16)
- 「マルチアドレス ファミリ VRF 構造」 (P.16)

EVN の利点

Easy Virtual Network は、レイヤ 3 ネットワークでのエンドツーエンドの仮想化を実現する IP ベースの仮想化テクノロジーです。ネットワーク仮想化は、ネットワークのセキュリティを保護するツールとして、また複数の仮想ネットワークに同じネットワーク インフラストラクチャを使用することで、ネットワークの資本と運用費を削減する方法として使用できます。それぞれ固有の論理ネットワークと一意のルーティングおよびフォワーディング テーブルを使用して、複数のグループをサポートすることにより、お客様は、同じ物理インフラストラクチャを何回でも利用できます。

ネットワーク仮想化以前、お客様は次の方法でパスの分離を実現していました。

- グループごとに専用ルータを使用して、物理的に分離します。これは、仮想ネットワークよりも費用がかかります。
- Access Control List (ACL; アクセス コントロール リスト) を使用しますが、ACL は一意のルーティングおよびフォワーディング テーブルをサポートしておらず、維持に費用がかかり、仮想ネットワークより、エラーが起こりやすい可能性があります。

EVN には次の利点があります。

- トラフィックを分離するために個別の物理インフラストラクチャを維持する必要がないため、資本支出を削減します。1 つの IP ネットワークで、ネットワークのトラフィックのパスを分離する複数の仮想ネットワークを設定できるため、追加のハードウェアの費用を節約します。
- 合併、買収、ビジネス パートナーのネットワーク統合が容易なため、ビジネスの柔軟性が高まります。
- ネットワークのコアを通るトラフィックの分離を維持するためのインフラストラクチャ要件が少ないため、複雑さを緩和します。
- Multi-VRF (VRF-Lite) と呼ばれる既存のメカニズムに基づいて構築されています。EVN は VRF-Lite と互換性があります。「EVN の VRF-Lite との互換性」 (P.16) を参照してください。EVN はパス分離の拡張機能、簡単な設定と管理、改善された共有サービスのサポートを提供するため、VRF-Lite よりも EVN が推奨されます。
- 社内の部署間でトラフィックの分割を維持することに加えて、パスの分離が役に立つ他のシナリオがあります。いくつかの例を次に示します。
 - インターネットへのゲストアクセス：お客様のネットワークを通るあらかじめ定められたデータ パスを使用し、ゲスト インターネット バウンド トラフィック用に一意のデフォルトのルートを定義できることで、インターネットへのゲストのネットワーク アクセスを制限します。

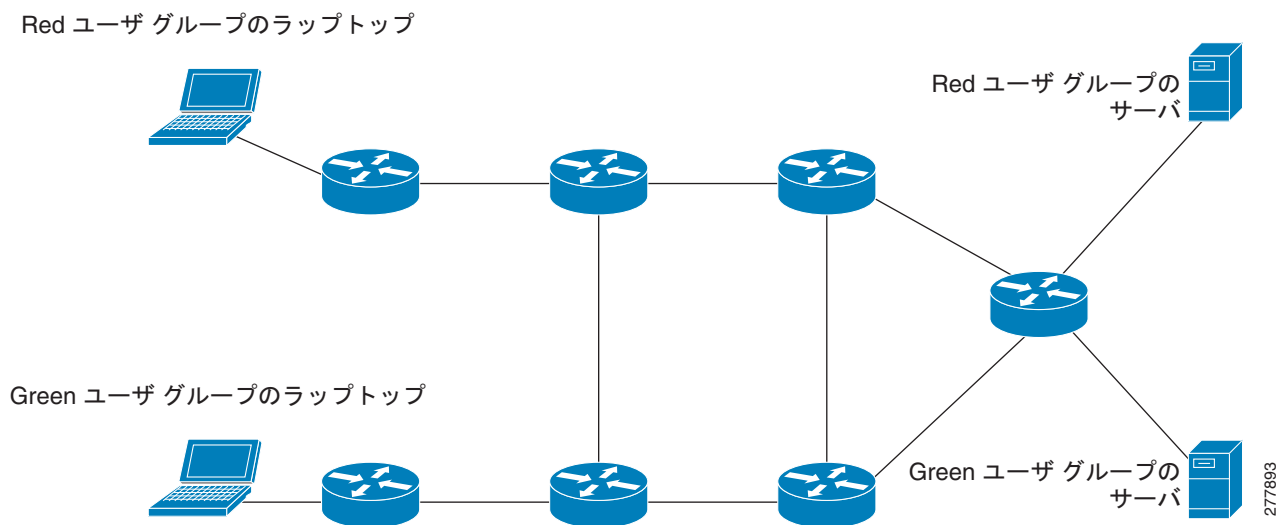
- ネットワーク アドミッション コントロール (NAC) の分離：非準拠のデスクトップから発信されたトラフィックを分離します。
- パートナー アクセス：お客様が許可するインターネット、電子メール、DNS、DHCP、またはアプリケーション サーバなどのネットワーク共有サービスへのパートナーや請負業者のアクセスを制限します。
- アプリケーションとデバイスの分離：トラフィックが検査される中央のファイアウォールにトラフィックを「強制的に通す」ことによって、データセンターのサービスとデバイスのセキュリティを保護します。
- アウトソーシング サービス：さまざまなクライアントからのデータ トラフィックを相互に分離します。
- スケーラブルなネットワーク：ネットワークの一部を、きわめて厳格なサービス レベルを必要とするトラフィックに制限し、それにより、必要な場所のみそれらの要件を提供することで、コストを低減することができます。
- 子会社/合併/買収：必要に応じてサービスの共有を可能にすると同時に、段階を追って会社やネットワークを統合します。
- サービス プロバイダーとしての役割を持つ企業：それぞれ 1 つの権限下で個別のネットワークを必要とする自立したグループ。例は、航空路線ごとに仮想ネットワークをサポートする空港です。

仮想ネットワーク タグによるパスの分離の提供

異なるユーザ グループを同じ IP インフラストラクチャで実行することは珍しくありません。さまざまなビジネス上の理由から、異なるグループ間でのトラフィックの分離が必要になります。図 1 は、同じネットワークで実行する Red と Green の 2 つのユーザ グループを示しています。ネットワーク仮想化の前は、2 つのグループ間でのトラフィックの分離はありません。Red ユーザ グループのユーザは、Green ユーザ グループのサーバにアクセスでき、逆も可能です。

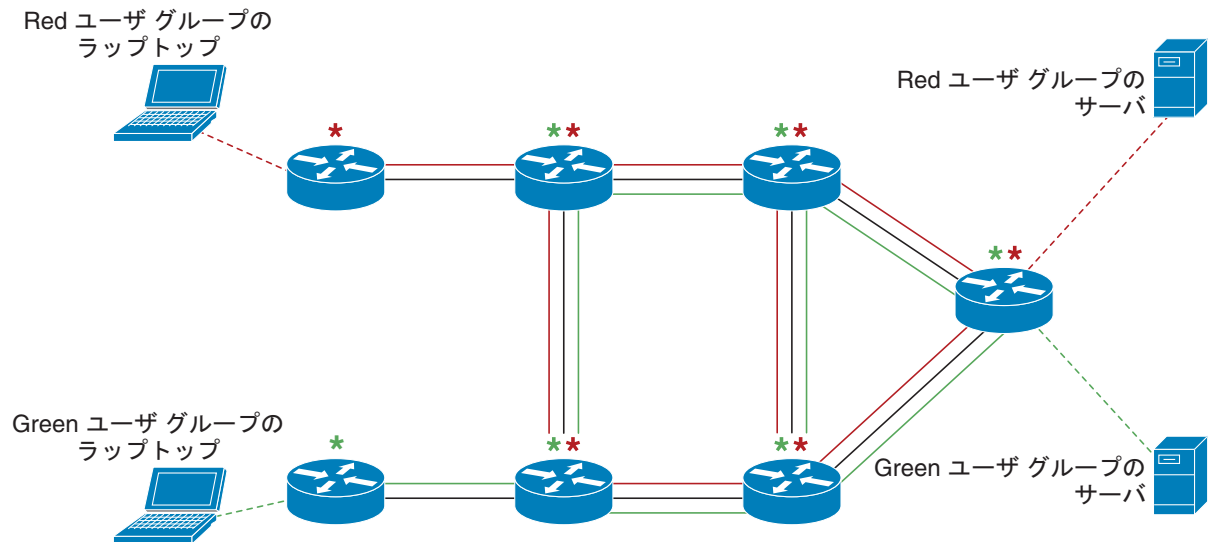
ネットワーク仮想化を使用しない場合、パスの分離はアクセス コントロールによって実現できますが、これは、維持に費用がかかり、エラーが起きやすく、ネットワークごとに一意のルーティングおよびフォワーディング テーブルをサポートしていません。

図 1 仮想化しないネットワーク



仮想ネットワークは、1つの物理ネットワーク上のさまざまなユーザグループを大まかにセグメント化します。仮想ネットワークを設定することによって、単一のIPインフラストラクチャを仮想化して、多数のエンドツーエンドの仮想ネットワークを実現できます。図2では、RedとGreenの2つのVRFを作成して、単一のIPインフラストラクチャを2つのVirtual Private Network (VPN; バーチャルプライベートネットワーク)に仮想化しています。

図 2 仮想化したネットワーク



- * * 各アスタリスクは VRF 定義を示す
- - - 各点線はエッジ インターフェイスを示す
- 実線の平行線グループはトランク インターフェイスを示す

277894

VRF を使用して、デバイスレベルの分割を提供するだけでなく、各仮想ネットワークは相互にパスも分離されます。パスの分離は、トラフィックを明示的にタグ付けすることによって実現されます。トラフィックは同じ仮想ネットワーク全体で同じタグ値を伝送します。パスに沿った各ネットワーク デバイスは、タグを使用して、異なる VRF 間で分割を行います。1 つのタグ番号は、たとえば、あるルータ上の VRF red を別のルータ上の VRF red に関連付けます。

仮想ネットワーク タグ

各 VPN と関連付けられた EVN は、設定時に割り当てるタグ値を持ちます。タグ値はグローバルです。つまり、各ルータで同じ EVN には同じ数値のタグ値が関連付けられる必要があります。タグ値の範囲は、2 ~ 4094 です。EVN は、ファスト イーサネットやギガビット イーサネットなどの 802.1q カプセル化をサポートするすべてのインターフェイスで使用できます。VRF-Lite ソリューションとの下位互換性を可能にするため、802.1q フレームの vLAN ID フィールドを使用して、仮想ネットワーク タグが伝送されます。

仮想ネットワーク タグを伝送するトラフィックはタグ付きトラフィックと呼ばれます。仮想ネットワーク タグを伝送しないトラフィックはタグなしトラフィックと呼ばれます。

red と green の 2 つの VRF がある次の構成で、タグを説明します。2 つの VRF はそれぞれ固有のタグを持つため、ルータのタグの構成は次のように見えます。

```
! Define two VRFs, red and green.
vrf definition red
  vnet tag 101
!
address-family ipv4
```



```
exit-address-family
!
vrf definition green
  vnet tag 102
!
  address-family ipv4
  exit-address-family
!
```

上に示すように、仮想ネットワークは、仮想ネットワーク タグが割り当てられている VRF インスタンスとして定義されます。

vnet Global

ルータ上に「vnet global」と呼ばれる事前定義された EVN があります。これは、グローバル ルーティング コンテキストを表し、デフォルトの RIB に対応します。図 2 と図 3 では、vnet global がルータを接続する黒線で表されています。vnet global はタグなしトラフィックを伝送します。デフォルトで、インターフェイスは vnet global に属します。さらに、vnet global は常にトランク インターフェイス上で実行します。vnet global はデフォルトのルーティング テーブルとも呼ばれます。

IPv6 トラフィックは vnet global でのみサポートされます。

エッジ インターフェイスと EVN トランク インターフェイス

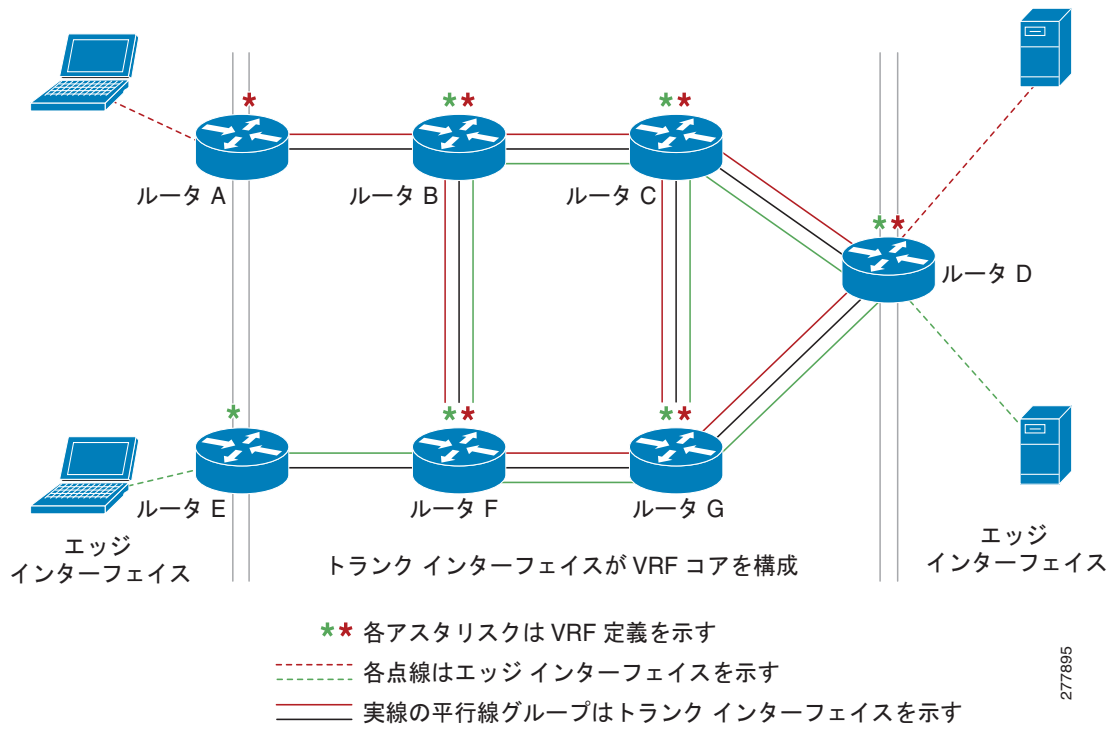
ユーザ デバイスはレイヤ 2 スイッチ ポートに接続され、VLAN に割り当てられています。VLAN はレイヤ 2 VPN として考えることができます。お客様は、単一の VLAN の共通の L3VPN でサポートされる必要があるすべてのデバイスをグループ化します。VLAN と VRF 間でデータ トラフィックが処理されるポイントはエッジインターフェイスと呼ばれます。

- エッジ インターフェイスはユーザ デバイスを EVN に接続し、事実上、EVN の境界を定義します。エッジ インターフェイスは VRF 対応でないホストやサーバなどのデバイスを接続します。エッジ インターフェイス経由で伝送されるトラフィックはタグが付けられません。エッジ インターフェイスは、受信したトラフィックが属する EVN を分類します。各エッジ インターフェイスは、1 つだけの EVN に属するように設定されます。
- EVN トランク インターフェイスは、VRF 対応ルータ同士を接続し、コアに、複数の EVN のトラフィックを転送するための手段を提供します。トランク インターフェイスはタグ付きトラフィックを伝送します。タグは、対応する EVN へのパケットの多重化を解除するために使われます。トランク インターフェイスには、EVN ごとに 1 つずつサブインターフェイスがあります。インターフェイスは **vnet trunk** コマンドによって、EVN トランク インターフェイスとして定義されます。

EVN インターフェイスは、エッジ インターフェイスとトランク インターフェイスの 2 種類のインターフェイスを使用します。インターフェイスはエッジ インターフェイスまたはトランク インターフェイスのいずれかで、両方にはなりません。図 3 に VRF Red に属するエッジ インターフェイスがある Router A と D を示します。Router D および E には VRF Green に属するエッジ インターフェイスがあります。

Router B、C、D、F、G には EVN コアを構成するトランク インターフェイスがあります。これらの 5 台のルータは VRF Red と VRF Green の両方に属するインターフェイスがあります。

図 3 EVN のエッジインターフェイスとトランク インターフェイス



仮想ネットワーク サブインターフェイスと出力でトランク インターフェイスを識別するための表記

トランク インターフェイスは複数の EVN を伝送するため、トランク インターフェイス名を表示するだけでは十分でない場合があります。出力がトランク インターフェイス上で実行する特定の EVN に含まれることを示す必要がある場合、使用する表記は、ピリオドと仮想ネットワーク タグを追加して、`<interface>.<virtual-network-tag>` の形式になります。例は、`gigabitethernet1/1/1.101` や `gigabitethernet1/1/1.102` のようになります。

デフォルトで、トランク インターフェイスが設定されると、すべての EVN と関連付けられた仮想ネットワーク タグが設定され、仮想ネットワーク サブインターフェイスが自動的に作成されます。上記のように、ピリオドと仮想ネットワーク タグ番号がインターフェイス番号に追加されます。

次の例では、VRF red が仮想ネットワーク タグ 3 で定義されています。従って、システムにより、ファストイーサネット 0/0/0.3 (VRF red の) が作成されました。

```
Router# show running-config vrf red
```

```
Building configuration...
```

```
Current configuration : 1072 bytes
vrf definition red
 vnet tag 3
 !
 address-family ipv4
 exit-address-family
 !
```

この非表示インターフェイスは **show derived-config** コマンドで表示でき、ファストイーサネット 0/0/0 に入力されたすべてのコマンドが、ファストイーサネット 0/0/0.3 に継承されていることがわかります。

```
Router# show derived-config interface fastethernet0/0/0.3
```

```
Derived configuration : 478 bytes
!
interface FastEthernet0/0/0.3
  description Subinterface for VRF NG red
  vrf forwarding red
  encapsulation dot1Q 3
  ip address 10.1.1.1 255.255.255.0
  ip authentication mode eigrp 1 md5
  ip authentication key-chain eigrp 1 x
  ip bandwidth-percent eigrp 1 3
  ip hello-interval eigrp 1 6
  ip hold-time eigrp 1 18
  no ip next-hop-self eigrp 1
  no ip split-horizon eigrp 1
  ip summary-address eigrp 1 10.0.0.0 255.0.0.0
end
```

トランク インターフェイスの単一の IP アドレス

トランク インターフェイスは複数の EVN にトラフィックを伝送できます。設定プロセスを簡単にするため、すべてのサブインターフェイスと関連付けられた EVN には同じ IP アドレスが割り当てられます。言い換えると、トランク インターフェイスは、異なる EVN コンテキストで、同じ IP アドレスによって識別されます。これは、各 EVN が一意のルーティングおよびフォワーディング テーブルを持つため、複数の EVN で重複する IP アドレスをサポートできる結果として実現されています。

定義されている VRF とトランク インターフェイス上で実行する VRF 間の関係

デフォルトで、ルータ上のトランク インターフェイスは、**vrf definition** コマンドで定義されたすべての VRF にトラフィックを伝送します。たとえば、次の構成では、ルータに定義されたすべての VRF がこのインターフェイスに含まれています。

```
interface FastEthernet 1/0/0
  vnet trunk
  ip address 10.1.1.1 255.255.255.0
```

ただし、トラフィック分割の目的で、特定のトランク インターフェイス上で VRF のサブセットのみを有効にしたい場合があります。これを実現するには、VRF リストを作成します。このリストは **vnet trunk** コマンドで参照されます。VRF リストでトランク インターフェイスを有効にすると、リスト上の VRF のみがインターフェイスで有効になります。例外として、**vnet global** はトランク インターフェイスで常に有効にされます。

次の例では、リストに指定された 2 つの VRF (red と green) のみがこのインターフェイスで有効にされます。

```
vrf list mylist
  member red
  member green
!
```

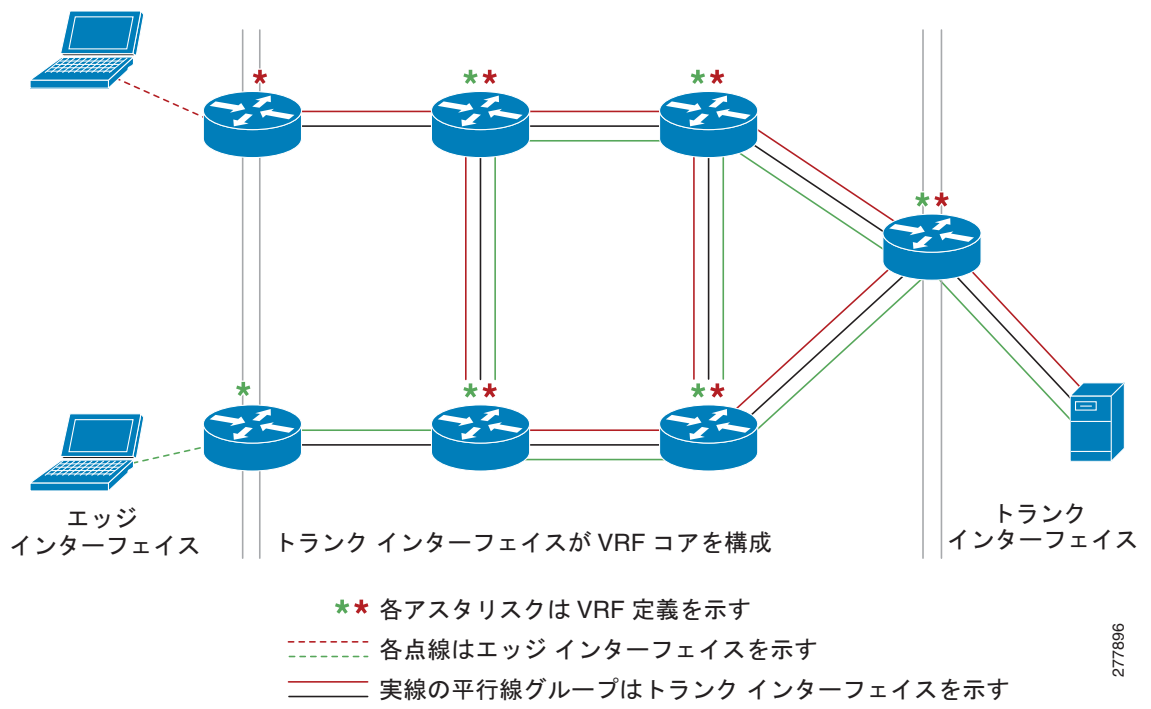
```
interface FastEthernet 1/0/0
vnet trunk list mylist
ip address 10.1.1.1 255.255.255.0
```

VRF 認識

仮想ネットワークに接続されたデバイスが仮想ネットワーク タグを理解できず、タグなしトラフィックしか送受信できないことがあります。そのようなデバイスは VRF 認識不能と呼ばれます。たとえば、ラップトップ コンピュータは通常 VRF 認識不能です。

逆に、タグ付きトラフィックを送受信でき、そうしたトラフィックを処理する時にタグ値を考慮するデバイスは、VRF 認識と呼ばれます。たとえば、異なる EVN 間で共有されている VRF 対応サーバは、仮想ネットワーク タグを使用して、受信した要求を区別し、それに従って応答を送信します。VRF 認識デバイスは、図 4 に示すように、トランク インターフェイスを使用して、EVN に接続されます。

図 4 VRF 認識サーバ



「VRF 認識」という用語は、ルータで実行するソフトウェア コンポーネントを説明するためにも使うことができます。ソフトウェア コンポーネントは異なる EVN 上で動作できる場合、VRF 認識です。たとえば、ping は、ping パケットを送信する EVN を選択できるため、VRF 認識です。

EVN でサポートされるルーティング プロトコル

各 EVN はルーティング プロトコルの個別のインスタンスを実行します。これにより、EVN ごとにルーティングを個別に微調整することが可能になり、運命共同体的な事態の発生を抑えることができます。異なる仮想ネットワークで異なるルーティング プロトコルを同時に実行できます。

EVN は、ユニキャスト ルーティングの場合に、スタティック ルート、OSPFv2、および EIGRP をサポートし、マルチキャスト ルーティングの場合に、PIM、MSDP、および IGMP をサポートします。

仮想ネットワークの packets フロー

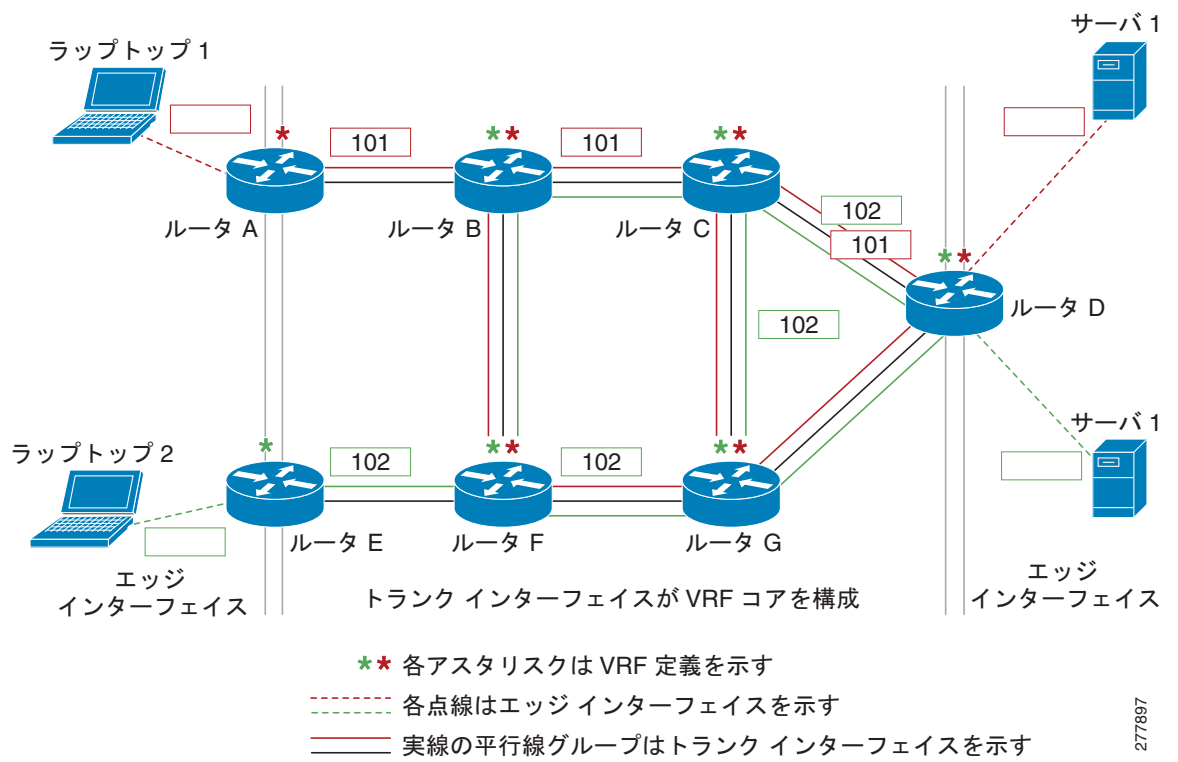
パケットは、エッジ インターフェイスを通過して EVN に入り、複数の トランク インターフェイスを通過して、別のエッジ インターフェイスから仮想ネットワークを出ます。入力エッジ インターフェイスでは、パケットが VLAN から特定の EVN にマッピングされます。パケットは、EVN にマッピングされると、関連付けられた仮想ネットワーク タグでタグ付けされます。仮想ネットワーク タグにより、トランク インターフェイスは複数の EVN にパケットを送信できます。パケットは、出力エッジ インターフェイスを通過して EVN を出るまで、タグが付けられたままになります。

エッジ インターフェイスでは、インターフェイスに関連付けられている EVN がルートのルックアップで使用されます。トランク インターフェイスで、パケットで送信された仮想ネットワーク タグが使われ、パケットをルーティングするための対応する EVN が検索されます。

出力インターフェイスがエッジ インターフェイスの場合、パケットはタグが外されて転送されます。ただし、出力インターフェイスがトランク インターフェイスの場合、パケットは入力 EVN のタグが付けられて転送されます。

図 5 に、red と green の 2 つの VRF からトラフィックが、タグ 101 と 102 を使用して、同じ IP インフラストラクチャ上で共存できる状況を示します。

図 5 仮想ネットワークの packets フロー



VRF red の Laptop 1 から Server 1 への packets フローは次のようになります。

1. Laptop 1 がタグなし packets を Server 1 に送信します。

2. Router A はエッジ インターフェイス上でパケットを受信し、これは VRF red に関連付けられています。
 - a. Router A は VRF red でルート ルックアップを実行し、次のホップがトランク インターフェイスから Router B であることがわかります。
 - b. Router A は VRF red のタグ (101) でパケットをカプセル化し、それをトランク インターフェイス上で送信します。
3. Router B はトランク インターフェイス上でパケットを受信します。Router B は仮想ネットワーク タグ 101 を確認して、そのパケットが VRF red に属することを識別します。
 - a. Router B は VRF red でルート ルックアップを実行し、次のホップがトランク インターフェイスから Router C であることがわかります。
 - b. Router B は VRF red のタグ (101) でパケットをカプセル化し、それをトランク インターフェイス上で送信します。
4. Router C はトランク インターフェイス上でパケットを受信します。Router C は仮想ネットワーク タグ 101 を使用して、そのパケットが VRF red に属することを識別します。
 - a. Router C は VRF red でルート ルックアップを実行し、次のホップがトランク インターフェイスから Router D であることがわかります。
 - b. Router C は VRF red のタグ (101) でパケットをカプセル化し、それをトランク インターフェイス上で送信します。
5. Router D はトランク インターフェイス上でパケットを受信します。Router D は仮想ネットワーク タグ 101 を使用して、そのパケットが VRF red に属することを識別します。
 - a. Router D は VRF red でルート ルックアップを実行し、次のホップがエッジ インターフェイスからわかります。
 - b. Router D は、エッジ インターフェイス上でタグなしパケットを Server 1 に送信します。
6. Server 1 は Laptop 1 から発信されたタグなしパケットを受信します。

EVN トランク インターフェイスでのコマンド継承

EVN トランク インターフェイスは、別々の EVN の設定要件が単一のトランク インターフェイス上で類似するという事実を利用します。トランク インターフェイスに設定された適格なコマンドは、同じインターフェイス上で実行するすべての EVN に継承されます。たとえば、OSPF hello interval は、次のように 1 行の設定で、トランク インターフェイス上のすべての EVN に設定できます。

```
interface gigabitethernet1/1/1
  vnet trunk
  ip address 10.1.2.1 255.255.255.0
  ! set OSPF hello interval for all VRFs on this interface.
  ip ospf hello-interval 20
```

同じインターフェイス上で実行するすべての EVN によって継承可能な、トランク インターフェイス上に設定されるコマンドのリストについては、「[Configuring Easy Virtual Networks](#)」モジュールで説明しています。

コマンド継承のその他の例については、「[Configuring Easy Virtual Networks](#)」モジュールの設定例を参照してください。

コマンド継承の上書き：仮想ネットワーク インターフェイス モード

同じトランク インターフェイス上の一部の EVN の設定を変えたい場合があります。その場合、仮想ネットワーク インターフェイス モードを使用して、特定の EVN の設定を行い、これにより、継承を上書きします。特定のインターフェイス コマンド（すべてのコマンドではない）は、VRF 単位で上書きできます。

インターフェイス コンフィギュレーション モードから、**vnet name** コマンドを入力すると、システムが仮想ネットワーク インターフェイス モードになります。このモードのシステム プロンプトは **Router(config-if-vnet)#** です。

次の例では、VRF **blue** の **30** のコストでインターフェイス上の他の VRF の **20** のコストを上書きします。

```
interface gigabitethernet 2/0/0
  vnet trunk
  ip address 10.1.1.1 255.255.255.0
  ! Set OSPF cost for all VRFs on this interface to 20.
  ip ospf cost 20
vnet name blue
  description Subinterface for VRF NG blue
  ! Set OSPF cost for blue to 30.
  ip ospf cost 30
```

show derived コマンドで、**30** のコストに変更されたサブインターフェイスが示されます。

```
Router(config-if-vnet)# do show derived | s interface GigabitEthernet2/0/0

interface GigabitEthernet2/0/0
vnet trunk
ip address 10.1.1.1 255.255.255.0
ip ospf cost 20
interface GigabitEthernet2/0/0.200
description Subinterface for VRF NG blue
vrf forwarding blue
ip address 10.1.1.1 255.255.255.0
ip ospf cost 30
Router(config-if-vnet)#
```

vnet Global のみへのアトリビュートの適用

同様に、**vnet global** のみにアトリビュートを有効にしたい場合があります。これを実行するには、次のように、**vnet global** インターフェイス サブモードを使用します。

```
interface gigabitethernet1/1/1
  vnet trunk
  ip address 10.1.2.1 255.255.255.0
  vnet global
  ! Set OSPF cost for global to 40.
  ip ospf cost 40
```

この例では、ユーザは **vnet global** を除くすべての EVN に EIGRP インターフェイス アトリビュートを設定しようとしています。**vnet global** を除いて、すべての EVN は 20 秒の保留時間を継承します。**vnet global** は 20 秒の保留時間が 40 秒で上書きされます。

```
interface fastethernet 1/0/0
  vnet trunk
  ip address 10.1.3.1 255.255.255.0
  ip hold-time eigrp 1 20
  vnet global
  ip hold-time eigrp 1 40
```

上書きの削除と継承される値の復元

no キーワードと **default** キーワードは、それらをトランク インターフェイスに使うか、仮想ネットワーク インターフェイス モードで使うかによって、結果が異なります。

- トランク インターフェイス上でコマンドの前に、**no** キーワードまたは **default** キーワードを入力すると、トランクがそのコマンドのシステムのデフォルト値に復元されます（これは、**no** または **default** キーワードの標準の動作結果です）。
- 仮想ネットワーク インターフェイス モードで、コマンドの前に **default** キーワードを入力すると、上書き値が削除され、トランクから継承された値が復元されます。特定の EVN の上書き値が無効になります。

次の例では、20 の OSPF コストでトランク インターフェイスが構成されますが、VRF blue により、その値が 30 の OSPF コストで上書きされます。

```
interface gigabitethernet 2/0/0
 vnet trunk
 ip address 10.1.1.1 255.255.255.0
 ! Set OSPF cost for all VRFs on this interface to 20.
 ip ospf cost 20
 vnet name blue
 ! Set OSPF cost for blue to 30.
 ip ospf cost 30
```

次のコマンドが入力された場合、OSPF コスト値が、トランク インターフェイスから継承されたコストである 20 に復元されます（20 は **ip ospf cost** コマンドのデフォルト値ではないことに注意してください）。

```
Router(config-if)# vnet name blue
Router(config-if-vnet)# default ip ospf cost
```

- 仮想ネットワーク インターフェイス モードでコマンドの前に **default** キーワードを入力すると、デフォルトの状態に復元されますが、**no** キーワードでは、必ずそうなるとは限りません。次の例で、**no ip dampening-change eigrp 1** はダンプニングの変更を無効にします。

```
interface Ethernet1/1
 vnet trunk
 ip dampening-change eigrp 1 50
 shutdown
 vnet name red
 no ip dampening-change eigrp 1
 ! Make sure vnet red does NOT have dampening change enabled, regardless of trunk setting.
 !
```

- コマンドが機能のオンとオフを切り替える種類である場合、コマンドの **no** 形式がコンフィギュレーション ファイルに表示されます。つまり、NVGEN 処理されて、トランクからの設定が上書きされます。次に例を示します。

```
interface gigabitethernet 2/0/0
 vnet trunk
 ip access-group 1 in
 vnet name red
 no ip pim sparse-mode
 no ip route-cache cef
 no ip access-group in
 vnet global
 ip ospf cost 100
```


- コマンドがその構文で **ip ospf cost cost** などの引数をとる場合、コマンドの **no** 形式によって、設定が削除されますが、コンフィギュレーション ファイルには表示されません。つまり、ユーザは **ip ospf cost default-value** を入力して、より直接的な方法で継承された値を上書きできるため、NVGEN 処理されません。

EXEC ルーティング コンテキスト

複数の EXEC コマンドを発行して、単一の EVN に適用したい場合があります。複数の EXEC コマンドに VRF 名を入力する回数を減らすため、**routing-context vrf** コマンドを使用して、それらの EXEC コマンドの VRF コンテキストを 1 回設定してから、EXEC コマンドに進みます。

表 1 は、ルーティング コンテキストの有無で、Cisco IOS の 4 つの EXEC コマンドを対比させています。左の列では、各 EXEC コマンドで VRF を識別する必要があることに注意してください。右の列では、VRF コンテンツが 1 回で識別され、プロンプトがその VRF を反映して変更されるため、コマンドごとに VRF を識別する必要はありません。

表 1 ルーティング コンテキストでの EXEC コマンド

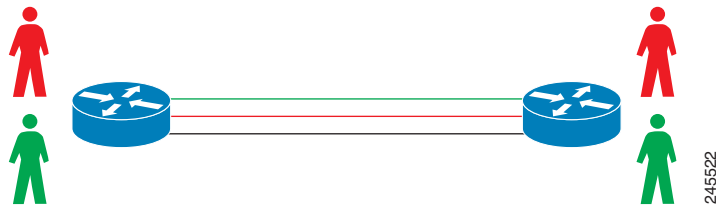
ルーティング コンテキストのない Cisco IOS CLI	ルーティング コンテキスト
—	Router# routing-context vrf red Router%red#
Router# show ip route vrf red [VRF red のルーティング テーブル出力]	Router%red# show ip route [VRF red のルーティング テーブル出力]
Router# ping vrf red 10.1.1.1 [VRF red を使用した ping 結果]	Router%red# ping 10.1.1.1 [VRF red を使用した ping 結果]
Router# telnet 10.1.1.1 /vrf red [VRF red の 10.1.1.1 への Telnet]	Router%red# telnet 10.1.1.1 [VRF red の 10.1.1.1 への Telnet]
Router# traceroute vrf red 10.1.1.1 [VRF red の Traceroute 出力]	Router%red# traceroute 10.1.1.1 [VRF red の Traceroute 出力]

EVN の VRF-Lite との互換性

EVN は VRF-Lite と密接な互換性があります。言い換えると、ボックス以外で、802.1q の使用、SNMP MIB、およびすべてのインフラストラクチャが VRF-Lite とまったく同じように見えます。

図 6 では、両方のルータに VRF が定義されています。左側のルータは VRF-Lite を使用し、右側のルータはタグ付きの EVN トランクを使用します。2 つの構成を図に示します。

図 6 VRF-Lite と EVN トランクの互換性



VRF-Lite サブインターフェイスの設定

```
interface TenGigabitEthernet1/1/1
ip address 10.122.5.31 255.255.255.254 vnet trunk
ip pim query-interval 333 msec
ip pim sparse-mode pim sparse-mode
logging event link-status logging event link-status

interface TenGigabitEthernet1/1/1.101
description Subinterface for Red VRF
encapsulation dot1Q 101 vnet tag 101
ip vrf forwarding Red
ip address 10.122.5.31 255.255.255.254
ip pim query-interval 333 msec vnet tag 102
ip pim sparse-mode
logging event subif-link-status

interface TenGigabitEthernet1/1/1.102
description Subinterface for Green VRF
encapsulation dot1Q 102
ip vrf forwarding Green
ip address 10.122.5.31 255.255.255.254
ip pim query-interval 333 msec
ip pim sparse-mode
logging event subif-link-status
```

EVN トランクの設定

マルチアドレス ファミリ VRF 構造

Cisco IOS Release 12.2(33)SB および 15.0(1)M 以前では、VRF の CLI は一度に 1 つのアドレス ファミリにのみ適用できました。たとえば、次のコマンドは IPv4 アドレス ファミリにのみ適用されます。

```
ip vrf blue
```

Cisco IOS Release 12.2(33)SB および 15.0(1)M では、VRF の CLI は同じ VRF 下の複数のアドレスファミリに適用されます。これは、マルチプロトコル VRF と呼ばれます。たとえば、次のコマンドは IPv4 VPN と IPv6 VPN に同時に適用されます。2 つのプロトコルのルーティング テーブルは異なります。

```
vrf definition blue
```



(注) Cisco IOS XE Release 3.2S で、仮想ネットワークは、**vnet global** を除いて、IPv6 をサポートしません。

参考資料

関連資料

関連項目	参照先
Cisco IOS コマンド	『 Cisco IOS Master Commands List, All Releases 』
Easy Virtual Network コマンド	『 Easy Virtual Network Command Reference 』
Easy Virtual Network の設定	「 Configuring Easy Virtual Network 」
Easy Virtual Network 共有サービスとルート レプリケーションの設定	「 Configuring Easy Virtual Network Shared Services 」
Easy Virtual Network の管理とトラブルシューティング	「 Easy Virtual Network Management and Troubleshooting 」

MIB

MIB	MIB リンク
VRF 情報を提供する MIB は引き続き Easy Virtual Network で動作します。システムのすべての VRF に関する VRF 独立 MIB のレポート情報： <ul style="list-style-type: none"> • CISCO—VRF-MIB • CISCO—MVPN-MIB • MPLS-VPN MIB 	選択したプラットフォーム、Cisco ソフトウェア リリース、および機能セットの MIB を検索してダウンロードする場合は、次の URL にある Cisco MIB Locator を使用します。 http://www.cisco.com/go/mibs

シスコのテクニカル サポート

説明	リンク
<p>右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。</p> <p>以下を含むさまざまな作業にこの Web サイトが役立ちます。</p> <ul style="list-style-type: none">・テクニカル サポートを受ける・ソフトウェアをダウンロードする・セキュリティの脆弱性を報告する、またはシスコ製品のセキュリティ問題に対する支援を受ける・ツールおよびリソースへアクセスする<ul style="list-style-type: none">- Product Alert の受信登録- Field Notice の受信登録- Bug Toolkit を使用した既知の問題の検索・Networking Professionals (NetPro) コミュニティで、技術関連のディスカッションに参加する・トレーニング リソースへアクセスする・TAC Case Collection ツールを使用して、ハードウェアや設定、パフォーマンスに関する一般的な問題をインタラクティブに特定および解決する <p>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。</p>	<p>http://www.cisco.com/cisco/web/support/index.html</p>

Easy Virtual Network の概要の機能情報

表 2 に、この機能のリリース履歴を示します。

表 2 に、このモジュールに記載されている機能および具体的な設定情報へのリンクを示します。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェア イメージのサポート情報を検索できます。Cisco Feature Navigator を使用すると、ソフトウェア イメージがサポートする特定のソフトウェア リリース、機能セット、またはプラットフォームを確認できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。Cisco.com のアカウントは必要ありません。



(注) 表 2 には、一連のソフトウェア リリースのうち、特定の機能が初めて導入されたソフトウェア リリースだけが記載されています。特に明記していないかぎり、その機能は、一連のソフトウェア リリースの以降のリリースでもサポートされます。

表 2 Easy Virtual Network の概要の機能情報

機能名	リリース	機能情報
Easy Virtual Network トランク	Cisco IOS Release XE 3.2S	Easy Virtual Network は、ネットワークのエンドツーエンドの仮想化を実現する IP ベースの仮想化テクノロジーです。単一の IP インフラストラクチャを使用して、トラフィック パスが相互に独立した状態で、個別の仮想ネットワークを提供できます。

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at www.cisco.com/go/trademarks. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1005R)

このマニュアルで使用している IP アドレスおよび電話番号は、実際のアドレスおよび電話番号を示すものではありません。マニュアル内の例、コマンド出力、ネットワーク トポロジ図、およびその他の図は、説明のみを目的として使用されています。説明の中に実際のアドレスおよび電話番号が使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

© 2010 Cisco Systems, Inc.
All rights reserved.

Copyright © 2010–2011, シスコシステムズ合同会社.
All rights reserved.



Easy Virtual Network の設定

Easy Virtual Network (EVN) は、ネットワークのエンドツーエンドの仮想化を実現する IP ベースの仮想化テクノロジーです。単一の IP インフラストラクチャを使用して、トラフィック パスが相互に独立した状態で、個別の仮想ネットワークを提供できます。Easy Virtual Network を設定して、複数の仮想 IP ネットワークを設定します。

機能情報の検索

ご使用のソフトウェア リリースによっては、このモジュールに記載されている機能の中に、一部サポートされていないものがあります。最新の機能情報と注意事項については、ご使用のプラットフォームとソフトウェア リリースに対応したリリース 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、および各機能がサポートされているリリースのリストについては、「[Easy Virtual Network の設定の機能情報](#)」(P.21) を参照してください。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェア イメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。Cisco.com のアカウントは必要ありません。

目次

- 「EVN の設定に関する前提条件」(P.2)
- 「EVN の設定に関する情報」(P.2)
- 「EVN の設定方法」(P.5)
- 「EVN の設定例」(P.13)
- 「その他の参考資料」(P.20)
- 「Easy Virtual Network の設定の機能情報」(P.21)

EVN の設定に関する前提条件

- ネットワークへの EVN の実装には、複数の仮想ネットワークを作成するために使用する単一の IP インフラストラクチャが必要です。異なる仮想ネットワーク上のトラフィックのパス分離が必要です。
- 「[Overview of Easy Virtual Network](#)」モジュールの概念を理解する必要があります。
- 各ルータのどのインターフェイスがどの EVN に属するのかわかるネットワーク トポロジを描くことをお勧めします。この図により、エッジ インターフェイスとして設定するインターフェイスと トランク インターフェイスとして設定するインターフェイスの追跡が容易になります。

EVN の設定に関する情報

- 「[仮想ネットワーク インターフェイス モード](#)」(P.2)

仮想ネットワーク インターフェイス モード

EVN の利点の 1 つは、共通のインターフェイスで複数の EVN を簡単に設定できることです。EVN に関連付けられる各インターフェイスを個別に設定する必要がありません。インターフェイス コンフィギュレーション モードで特定のコマンドを使用する場合、それらは、**vnet global** を含め、その共通インターフェイスを共有する各 EVN によって継承されたデフォルト値を定義します。これらの設定が、指定のインターフェイスを共有するすべての EVN で満足できる場合は、仮想ネットワーク インターフェイス モードでのそれらの機能の追加の設定は必要ありません。

または、これらの値は、各 EVN に対し、仮想ネットワーク インターフェイス モードでコマンドを指定することによって、選択して上書きできます。これらのコマンドのいずれかが、*仮想ネットワーク インターフェイス モード*と呼ばれるモードで指定されている場合、Cisco デフォルト値またはユーザがインターフェイス コンフィギュレーション レベルで設定した値が上書きされます。

インターフェイス コンフィギュレーション モードから、**vnet name** コマンドを入力すると、システムが仮想ネットワーク インターフェイス モードになります。このモードのシステム プロンプトは **Router(config-if-vnet)#** です。仮想ネットワーク インターフェイス モードでは、特定のコマンドのインターフェイス設定を上書きできるため、上書きを特定の EVN に適用できます。

仮想ネットワーク インターフェイス モードで使用可能な OSPF コマンド

仮想ネットワーク インターフェイス モードでは、次の OSPF コマンドを使用できます。

- **ip ospf process-id area**
- **ip ospf authentication**
- **ip ospf authentication-key**
- **ip ospf bfd**
- **ip ospf cost**
- **ip ospf database-filter**
- **ip ospf dead-interval**
- **ip ospf demand-circuit**
- **ip ospf flood-reduction**

- **ip ospf hello-interval**
- **ip ospf lls**
- **ip ospf message-digest-key**
- **ip ospf mtu-ignore**
- **ip ospf network**
- **ip ospf priority**
- **ip ospf resync-timeout**
- **ip ospf shutdown**
- **ip ospf transmit-delay**
- **ip ospf transmit-interval**
- **ip ospf ttl-security**

仮想ネットワーク インターフェイス モードで使用可能な EIGRP コマンド

仮想ネットワーク インターフェイス モードでは、次の EIGRP コマンドを使用できます。

- **ip authentication key-chain eigrp**
- **ip authentication mode eigrp**
- **ip bandwidth-percent eigrp**
- **ip dampening-change eigrp**
- **ip dampening-interval eigrp**
- **ip hello-interval eigrp**
- **ip hold-time eigrp**
- **ip next-hop-self eigrp**
- **ip split-horizon eigrp**
- **ip summary-address eigrp**

仮想ネットワーク インターフェイス モードでは、EIGRP のインターフェイスのコストを判断する方法に影響する 2 つの追加のコマンドを使用できます。

- **bandwidth** (インターフェイス)
- **delay** (インターフェイス)

仮想ネットワーク インターフェイス モードで使用可能な IP マルチキャスト コマンド

仮想ネットワーク インターフェイス モードでは、次の IP マルチキャスト コマンドを使用できます。

- **ip igmp access-group**
- **ip igmp explicit-tracking**
- **ip igmp helper-address**
- **ip igmp immediate-leave**
- **ip igmp join-group**
- **ip igmp last-member-query-count**

- **ip igmp last-member-query-interval**
- **ip igmp limit**
- **ip igmp mroute-proxy**
- **ip igmp proxy-service**
- **ip igmp querier-timeout**
- **ip igmp query-interval**
- **ip igmp query-max-response-time**
- **ip igmp static-group**
- **ip igmp tcn**
- **ip igmp unidirectional-link**
- **ip igmp v3lite**
- **ip igmp version**
- **ip multicast boundary**
- **ip multicast helper map**
- **ip multicast limit**
- **ip pim bidir-neighbor-filter**
- **ip pim bsr-border**
- **ip pim dense-mode**
- **ip pim dr-priority**
- **ip pim nbma-mode**
- **ip pim neighbor-filter**
- **ip pim passive**
- **ip pim query-interval**
- **ip pim sparse-dense-mode**
- **ip pim sparse-mode**
- **ip pim state-refresh**
- **ip pim version**

仮想ネットワーク インターフェイス モードで使用可能なマルチキャスト転送情報ベース コマンド

仮想ネットワーク インターフェイス モードでは、次のマルチキャスト転送情報ベース (MFIB) コマンドを使用できます。

- **ip mfib cef**
- **ip mfib forwarding**

EVN の設定方法

複数の仮想ネットワークを設定するには、次のタスクを実行します。

- 「Easy Virtual Network トランク インターフェイスの設定」 (P.5) (必須)
- 「トランク インターフェイス上で VRF のサブセットのみを有効にする」 (P.9) (任意)
- 「エッジ インターフェイスの設定」 (P.11) (必須)
- 「EVN 設定の確認」 (P.12) (任意)

Easy Virtual Network トランク インターフェイスの設定

このタスクを実行して、EVN トランク インターフェイスを設定します。このインターフェイスはルータを接続して、複数の仮想ネットワークにトラフィックを転送するためのコアを提供します。トランク インターフェイス経由で伝送されるトラフィックにはタグが付けられます。次のタスクは、ベース VRF と、VRF red と VRF blue の 2 つの名前付き VRF があるトランク インターフェイスの設定方法を説明しています。

手順の概要

1. **enable**
2. **configure terminal**
3. **vrf definition** *vrf-name*
4. **vnet tag** *number*
5. **description** *string*
6. **address-family** *ipv4*
7. **exit-address-family**
8. **exit**
9. **vrf definition** *vrf-name*
10. **vnet tag** *number*
11. **description** *string*
12. **address-family** *ipv4*
13. **exit-address-family**
14. **exit**
15. **interface** *type number*
16. **ip address** *ip-address mask*
17. **vnet trunk** [**list** *vrf-list-name*]
18. **vnet name** *vrf-name*
19. **exit-if-vnet**
20. **no shutdown**
21. **exit**
22. **router ospf** *process-id*

23. `network ip-address wildcard area area-id`
24. `exit`
25. `router ospf process-id vrf vrf-name`
26. `network ip-address wildcard area area-id`
27. `exit`
28. `router ospf process-id vrf vrf-name`
29. `network ip-address wildcard area area-id`
30. `end`

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<code>enable</code> 例： Router> enable	特権 EXEC モードをイネーブルにします。 • 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	<code>configure terminal</code> 例： Router# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<code>vrf definition vrf-name</code> 例： Router(config)# vrf definition red	VRF ルーティング テーブル インスタンスを設定し、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	<code>vnet tag number</code> 例： Router(config-vrf)# vnet tag 100	VRF のグローバルな数値タグを指定します。 • 同じタグ番号を各エッジおよびトランク インターフェイス上の同じ仮想ネットワークに設定する必要があります。
ステップ 5	<code>description string</code> 例： Router(config-vrf) description guest access	(任意) コンフィギュレーション ファイルを参照するネットワーク管理者に役立つように、VRF について記述します。
ステップ 6	<code>address-family ipv4</code> 例： Router(config-vrf) address-family ipv4	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 アドレス プレフィ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 7	<code>exit-address-family</code> 例： Router(config-vrf-af) exit-address-family	アドレス ファミリ コンフィギュレーション モードを終了します。
ステップ 8	<code>exit</code> 例： Router(config-vrf) # exit	次に最高のコンフィギュレーション モードに移動します。

	コマンドまたはアクション	目的
ステップ 9	<code>vrf definition vrf-name</code> 例： Router(config)# vrf definition blue	VRF ルーティング テーブル インスタンスを設定し、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 10	<code>vnet tag number</code> 例： Router(config-vrf)# vnet tag 200	VRF のグローバルな数値タグを指定します。 • 同じタグ番号を各エッジおよびトランク インターフェイス上の同じ VRF に設定する必要があります。
ステップ 11	<code>description string</code> 例： Router(config-vrf) description Finance	(任意) コンフィギュレーション ファイルを参照するネットワーク管理者に役立つように、VRF について記述します。
ステップ 12	<code>address-family ipv4</code> 例： Router(config-vrf) address-family ipv4	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 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 13	<code>exit-address-family</code> 例： Router(config-vrf-af) exit-address-family	アドレス ファミリ コンフィギュレーション モードを終了します。
ステップ 14	<code>exit</code> 例： Router(config-vrf)# exit	次に最高のコンフィギュレーション モードに移動します。
ステップ 15	<code>interface type number</code> 例： Router(config)# interface gigabitethernet 1/1/1	インターフェイス タイプを設定します。
ステップ 16	<code>ip address ip-address mask</code> 例： Router(config-if)# ip address 10.1.1.1 255.255.255.0	インターフェイスのプライマリ アドレスを設定します。
ステップ 17	<code>vnet trunk [list vrf-list-name]</code> 例： Router(config-if)# vnet trunk	トランク インターフェイスを定義します。 • デフォルトで、 vrf definition コマンドによって定義されたすべての VRF がルータ上のすべてのトランク インターフェイスで実行します。そのため、VRF red と VRF blue は現在このインターフェイス上で実行しています。 • list vrf-list-name キーワード/引数のペアを使用して、このトランク インターフェイスで実行する VRF を制限します。

コマンドまたはアクション	目的
<p>ステップ 18 <code>vnet name vrf-name</code></p> <p>例： Router(config-if)# vnet name red</p>	<p>(任意) 仮想ネットワーク インターフェイス モードを開始します。</p> <ul style="list-style-type: none"> • vnet コマンドは、仮想ネットワーク インターフェイス モードを開始し、この VRF のみに適用する機能を設定して、グローバル値を上書きできます。インターフェイス上のすべての VRF についてグローバル設定で満足できる場合は、このステップは必要ありません。 • このステップの後、ip ospf cost などの適格なコマンドを設定できます (このタスクでは説明しません)。
<p>ステップ 19 <code>exit-if-vnet</code></p> <p>例： Router(config-if-vnet) exit-if-vnet</p>	<p>VRF インターフェイス コンフィギュレーション モードから、インターフェイス コンフィギュレーション モードに移動します。</p>
<p>ステップ 20 <code>no shutdown</code></p> <p>例： Router(config-if) no shutdown</p>	<p>インターフェイスを再起動します。</p>
<p>ステップ 21 <code>exit</code></p> <p>例： Router(config-if) exit</p>	<p>次に最高のコンフィギュレーション モードに移動します。</p>
<p>ステップ 22 <code>router ospf process-id</code></p> <p>例： Router(config)# router ospf 1</p>	<p>OSPF ルーティング プロセスを設定し、それを VRF に関連付けます。</p> <ul style="list-style-type: none"> • この OSPF インスタンスには VRF がないため、vnet global です。
<p>ステップ 23 <code>network ip-address wildcard area area-id</code></p> <p>例： Router(config-router) network 10.0.0.0 255.255.255.0 area 0</p>	<p>OSPF が実行するインターフェイスと、それらのインターフェイスに対するエリア ID を定義します。</p>
<p>ステップ 24 <code>exit</code></p> <p>例： Router(config-router) exit</p>	<p>次に最高のコンフィギュレーション モードに移動します。</p>
<p>ステップ 25 <code>router ospf process-id vrf vrf-name</code></p> <p>例： Router(config)# router ospf 2 vrf red</p>	<p>OSPF ルーティング プロセスを設定し、それを VRF に関連付けます。</p> <ul style="list-style-type: none"> • 各 VRF は固有の OSPF インスタンスが必要であるため、VRF ごとに異なる <i>process-id</i> を使用します。

	コマンドまたはアクション	目的
ステップ 26	network <i>ip-address wildcard area area-id</i> 例 : Router(config-router) network 10.0.0.0 255.255.255.0 area 0	OSPF が実行するインターフェイスと、それらのインターフェイスに対するエリア ID を定義します。
ステップ 27	exit 例 : Router(config-router) exit	次に最高のコンフィギュレーション モードに移動します。
ステップ 28	router ospf process-id vrf vrf-name 例 : Router(config)# router ospf 3 vrf blue	OSPF ルーティング プロセスを設定し、それを VRF に関連付けます。 <ul style="list-style-type: none"> 各 VRF は固有の OSPF インスタンスが必要であるため、VRF ごとに異なる <i>process-id</i> を使用します。
ステップ 29	network ip-address wildcard area area-id 例 : Router(config-router) network 10.0.0.0 255.255.255.0 area 2	OSPF が実行するインターフェイスと、それらのインターフェイスに対するエリア ID を定義します。
ステップ 30	end 例 : Router(config-vrf) end	設定セッションを終了し、特権 EXEC モードに戻ります。

トランク インターフェイス上で VRF のサブセットのみを有効にする

前のタスク「[Easy Virtual Network トランク インターフェイスの設定](#)」では、2 つの VRF でトランク インターフェイスを設定する方法を示しました。デフォルトで、ルータ上のトランク インターフェイスは、**vrf definition** コマンドで定義された各 VRF にトラフィックを伝送できます。ただし、トランク インターフェイス上で VRF のサブセットのみを有効にしたい場合があります。これは、VRF リストを作成して実行します。ルータには最大 32 の VRF リストが存在できます。VRF リストを作成するには、次の作業を実行します。このタスクは、VRF がすでに設定されていることを前提にしています。

手順の概要

1. **enable**
2. **configure terminal**
3. **vrf list vrf-list-name**
4. **member vrf-name**
5. 必要に応じて、ステップ 4 を繰り返して、その他の VRF をリストに追加します。
6. **exit-vrf-list**
7. **interface type number**
8. **vnet trunk list vrf-list-name**
9. **ip address ip-address mask**

10. end

11. show vrf list [vrf-list-name]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Router> enable	特権 EXEC モードをイネーブルにします。 • 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： Router# 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	vrf list vrf-list-name 例： Router(config)# vrf list External	VRF のリストを定義します。 • <i>vrf-list-name</i> には、最大 32 文字を含めることができません。疑問符、スペース、* は使用できません。
ステップ 4	member vrf-name 例： Router(config-vrf-list)# member blue	VRF リストのメンバとして既存の VRF を指定します。 • VRF は、リストに追加する前に、すでに定義されている必要があります。
ステップ 5	必要に応じて、ステップ 4 を繰り返して、その他の VRF をリストに追加します。	(任意) VRF が 1 つだけ必要なトランク インターフェイスにリストを適用する場合、VRF が 1 つだけのリストを作成することができます。
ステップ 6	exit-vrf-list 例： Router(config-vrf-list)# exit-vrf-list	VRF リスト コンフィギュレーション モードを終了します。
ステップ 7	interface type number 例： Router(config)# interface gigabitethernet 1/1/1	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。
ステップ 8	vnet trunk list vrf-list-name 例： Router(config-if)# vnet trunk list mylist	トランク インターフェイスを定義し、VRF リストにある VRF を有効にします。 • ステップ 3 で定義した <i>vrf-list-name</i> を使用します。
ステップ 9	ip address ip-address mask 例： Router(config-if)# ip address 10.1.3.1 255.255.255.0	インターフェイスのプライマリ アドレスを設定します。

	コマンドまたはアクション	目的
ステップ 10	<code>end</code> 例： Router(config-if) end	設定セッションを終了し、特権 EXEC モードに戻ります。
ステップ 11	<code>show vrf list [vrf-list-name]</code> 例： Router# show vrf list mylist	VRF リストの情報を表示します。

エッジ インターフェイスの設定


次のタスクを実行して、ユーザ デバイスを仮想ネットワークに接続するエッジ インターフェイスを設定します。エッジ インターフェイス経由で伝送されるトラフィックはタグが付けられません。エッジ インターフェイスは、受信したトラフィックが属する仮想ネットワークを判断します。各エッジ インターフェイスは、1 つだけの仮想ネットワークにマッピングされます。

手順の概要

1. `enable`
2. `configure terminal`
3. `interface type number`
4. `vrf forwarding vrf-name`
5. `ip address ip-address mask`
6. `end`

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<code>enable</code> 例： Router> enable	特権 EXEC モードをイネーブルにします。 • 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	<code>configure terminal</code> 例： Router# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<code>interface type number</code> 例： Router(config)# interface gigabitethernet 1/0/0	インターフェイス タイプを設定します。

コマンドまたはアクション	目的
<p>ステップ 4 <code>vrf forwarding vrf-name</code></p> <p>例 : Router(config-if)# vrf forwarding red</p>	<p>エッジ インターフェイスを定義し、着信トラフィックが属する VRF を判断します。</p> <ul style="list-style-type: none"> • vrf definition コマンドで <i>vrf-name</i> がすでに定義されている必要があります。 • この例では、着信トラフィックが VRF red に属しています。 <p> (注) エッジ インターフェイスを設定しようとする場合、トランク インターフェイス上にないことを確認してください。</p>
<p>ステップ 5 <code>ip address ip-address mask</code></p> <p>例 : Router(config-if)# ip address 10.1.1.1 255.255.255.0</p>	<p>インターフェイスのプライマリ アドレスを設定します。</p>
<p>ステップ 6 <code>end</code></p> <p>例 : Router(config-if) end</p>	<p>設定セッションを終了し、特権 EXEC モードに戻ります。</p>

この次の手順

エッジ インターフェイスとトランク インターフェイスを設定したら、ネットワーク図を参照して、別のルータにログオンします。エッジ インターフェイスがある場合、そのインターフェイスを設定します。トランク インターフェイスがある場合は、適切な VRF でそのインターフェイスを設定します。各 VRF に属する各ルータおよびインターフェイスの設定を続行します。

VRF で実行する他のプロトコル機能を設定します。該当する IP ルーティング設定ガイドを参照してください。

EVN 設定の確認

このタスクの次のいずれかのステップを実行して、設定を確認します。仮想ネットワークが VRF の場合、仮想ネットワークで既存のすべての VRF **show** コマンドがサポートされます。ルータに VRF と仮想ネットワークが混在する場合、各種 **show vrf** コマンドで、出力に、VRF と仮想ネットワークの両方が含まれます。

手順の概要

1. **enable**
2. **show vnet tag**
3. **show running-config [vrf | vnet] [vrf-name]**
4. **show vrf list [vrf-list-name]**
5. **show {vrf | vnet} [ipv4 | ipv6] [interface | brief | detail | lock] [vrf-name]**
6. **show {vrf | vnet} counters**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<code>enable</code> 例： Router> enable	特権 EXEC モードをイネーブルにします。 • 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	<code>show vnet tag</code> 例： Router# show vnet tag	(任意) 各タグが設定され、使われている場所を表示します。
ステップ 3	<code>show running-config [vrf vnet] [vrf-name]</code> 例： Router# show running-config vrf green	(任意) 実行コンフィギュレーションの VRF を表示し、VRF のインターフェイスを表示して、マルチ VRF のプロトコル設定を表示します。
ステップ 4	<code>show vrf list [vrf-list-name]</code> 例： Router# show vrf list	(任意) 各リスト上の VRF など、VRF リストに関する情報を表示します。
ステップ 5	<code>show {vrf vnet} [ipv4 ipv6] [interface brief detail lock] [vrf-name]</code> 例： Router# show vnet detail	(任意) VRF に関する情報を表示します。
ステップ 6	<code>show {vrf vnet} counters</code> 例： Router# show vnet counters	(任意) サポートされ、設定されている VRF または仮想ネットワークの数などの情報を表示します。

EVN の設定例

- 「例：network コマンドでの OSPF を使用した仮想ネットワーク」(P.13)
- 「例：ip ospf vnet area コマンドでの OSPF を使用した仮想ネットワーク」(P.14)
- 「例：EIGRP 環境でのコマンド継承と仮想ネットワーク インターフェイス モードの上書き」(P.15)
- 「例：マルチキャスト環境でのコマンド継承と仮想ネットワーク インターフェイス モードの上書き」(P.18)
- 「例：IP マルチキャストを使用した EVN」(P.19)

例：network コマンドでの OSPF を使用した仮想ネットワーク

この例では、**network** コマンドで共有 VRF インターフェイスをベース VRF と red および blue の 2 つの名前付き VRF に関連付けます。各 VRF には固有の OSPF インスタンスが必要であるため、3 つの OSPF インスタンスがあります。OSPF 1 には VRF がないため、これは **vnet global** です。

```

vrf definition red
vnet tag 100
address-family ipv4
exit-address-family
!
vrf definition blue
vnet tag 200
address-family ipv4
exit-address-family
!
interface gigabitethernet 0/0/0
ip address 10.0.0.1 255.255.255.0
vnet trunk
vnet name red
ip ospf cost 100
!
router ospf 1
log-adjacency-changes detail
network 10.0.0.0 255.255.255.0 area 0
router ospf 2 vrf red
log-adjacency-changes
network 10.0.0.0 255.255.255.0 area 0
router ospf 3 vrf blue
log-adjacency-changes
network 10.0.0.0 255.255.255.0 area 2

```

例 : ip ospf vnet area コマンドでの OSPF を使用した仮想ネットワーク

この例では、OSPF インスタンスと特定のインターフェイス間の関連付けに関して、前の例と異なります。この例では、OSPF がトランク インターフェイスのすべての仮想ネットワークで実行しています。**ip ospf vnet area** コマンドは GigabitEthernet 0/0/0 インターフェイスと 3 つの OSPF インスタンスを関連付けます。

```

vrf definition red
vnet tag 100
address-family ipv4
exit-address-family
!
vrf definition blue
vnet tag 200
address-family ipv4
exit-address-family
!
interface gigabitethernet 0/0/0
ip address 10.0.0.1 255.255.255.0
vnet trunk
ip ospf vnet area 0
vnet name red
ip ospf cost 100
vnet name blue
ip ospf 3 area 2
!
router ospf 1
log-adjacency-changes detail
router ospf 2 vrf red
log-adjacency-changes
router ospf 3 vrf blue
log-adjacency-changes

```

例：EIGRP 環境でのコマンド継承と仮想ネットワーク インターフェイスモードの上書き

GigabitEthernet インターフェイスがさまざまな EIGRP コマンドによって設定されているものとします。

```
interface gigabitethernet0/0/0
  vnet trunk
  ip address 1.1.1.1 255.255.255.0
  ip authentication mode eigrp 1 md5
  ip authentication key-chain eigrp 1 x
  ip bandwidth-percent eigrp 1 3
  ip dampening-change eigrp 1 30
  ip hello-interval eigrp 1 6
  ip hold-time eigrp 1 18
  no ip next-hop-self eigrp 1
  no ip split-horizon eigrp 1
  ip summary-address eigrp 1 1.0.0.0 255.0.0.0
end
```

トランクが設定されているため、VRF サブインターフェイスが自動的に作成され、メイン インターフェイスのコマンドが VRF サブインターフェイス (g0/0/0.3) に継承されます (3 は vnet tag 3 からのタグ番号です)。

```
R1# show running-config vrf red
```

```
Building configuration...
```

```
Current configuration : 1072 bytes
vrf definition red
  vnet tag 3
  !
  address-family ipv4
  exit-address-family
  !
```

show derived-config コマンドで非表示サブインターフェイスを表示し、GigabitEthernet 0/0/0 に入力されたすべてのコマンドが、GigabitEthernet 0/0/0.3 に継承されていることを確認します。

```
R1# show derived-config interface gigabitethernet0/0/0.3
```

```
Building configuration...
```

```
Derived configuration : 478 bytes
!
interface GigabitEthernet0/0/0.3
  description Subinterface for VNET red
  vrf forwarding red
  encapsulation dot1Q 3
  ip address 1.1.1.1 255.255.255.0
  ip authentication mode eigrp 1 md5
  ip authentication key-chain eigrp 1 x
  ip bandwidth-percent eigrp 1 3
  ip dampening-change eigrp 1 30
  ip hello-interval eigrp 1 6
  ip hold-time eigrp 1 18
  no ip next-hop-self eigrp 1
  no ip split-horizon eigrp 1
  ip summary-address eigrp 1 1.0.0.0 255.0.0.0
end
```

仮想ネットワーク インターフェイス モードを使用してそれらのコマンドを上書きできます (**vnet name** コマンドで)。次に例を示します。

```
R1(config)# interface gigabitethernet0/0/0
R1(config-if)# vnet name red

R1(config-if-vnet)# no ip authentication mode eigrp 1 md5 ! disable authen for e0/0.3 only
R1(config-if-vnet)# ip authentication key-chain eigrp 1 y ! different key-chain
R1(config-if-vnet)# ip bandwidth-percent eigrp 1 99 ! higher bandwidth-percent
R1(config-if-vnet)# no ip dampening-change eigrp 1 ! disable dampening-change
R1(config-if-vnet)# ip hello eigrp 1 7
R1(config-if-vnet)# ip hold eigrp 1 21
R1(config-if-vnet)# ip next-hop-self eigrp 1 ! enable next-hop-self for e0/0.3
R1(config-if-vnet)# ip split-horizon eigrp 1 ! enable split-horizon
R1(config-if-vnet)# no ip summary-address eigrp 1 1.0.0.0 255.0.0.0 ! do not summarize
on e0/0.3
```

```
R1(config-if-vnet)# do show running-config interface gigabitethernet0/0/0
```

Building configuration...

```
Current configuration : 731 bytes
!
interface GigabitEthernet0/0/0
 vnet trunk
 ip address 1.1.1.1 255.255.255.0
 ip authentication mode eigrp 1 md5
 ip authentication key-chain eigrp 1 x
 ip bandwidth-percent eigrp 1 3
 ip dampening-change eigrp 1 30
 ip hello-interval eigrp 1 6
 ip hold-time eigrp 1 18
 no ip next-hop-self eigrp 1
 no ip split-horizon eigrp 1
 ip summary-address eigrp 1 1.0.0.0 255.0.0.0
 vnet name red
 ip split-horizon eigrp 1
 no ip summary-address eigrp 1 1.0.0.0 255.0.0.0
 no ip authentication mode eigrp 1 md5
 ip authentication key-chain eigrp 1 y
 ip bandwidth-percent eigrp 1 99
 no ip dampening-change eigrp 1
 ip hello-interval eigrp 1 7
 ip hold-time eigrp 1 21
 ip next-hop-self eigrp 1
!
end
```

g0/0.3 で上書き設定が使われていることに注意します。

```
R1(config-if-vnet)# do show derived-config interface g0/0.3
Building configuration...
```

```
Derived configuration : 479 bytes
!
interface GigabitEthernet0/0/0.3
 description Subinterface for VNET red
 vrf forwarding red
 encapsulation dot1Q 3
 ip address 1.1.1.1 255.255.255.0
 no ip authentication mode eigrp 1 md5
 ip authentication key-chain eigrp 1 y
 ip bandwidth-percent eigrp 1 99
```

```

no ip dampening-change eigrp 1
ip hello-interval eigrp 1 7
ip hold-time eigrp 1 21
ip next-hop-self eigrp 1
ip split-horizon eigrp 1
no ip summary-address eigrp 1 1.0.0.0 255.0.0.0
end

```

vnet name サブモードで入力されたコマンドは、スティッキです。つまり、**vnet name** サブモードでコマンドを入力すると、それがデフォルト値と同じ値に設定されているかどうかに関係なく、NVGEN 処理されます。たとえばデフォルトの **hello** 値は 5 です。**vnet name** サブモードで **ip hello eigrp** コマンドを入力すると、それが NVGEN 処理されます。他のモードではそうなりません。

```

R1(config-if)# interface gigabitethernet0/0/2
R1(config-if)# vnet trunk
R1(config-if)# ip bandwidth-percent eigrp 1 50 <----<< this will NOT nvgen
R1(config-if)# ip hello eigrp 1 5 <----<< this will NOT nvgen
R1(config-if)# no ip authentication mode eigrp 1 md5 <---<< this will NOT nvgen
R1(config-if)# vnet name red
R1(config-if-vnet)# ip bandwidth-percent eigrp 1 50 <---<< this will nvgen
R1(config-if-vnet)# ip hello eigrp 1 5 <---<< this will nvgen
R1(config-if-vnet)# no ip authentication mode eigrp 1 md5 <---<< this will nvgen

R1(config-if-vnet)# do show running-config interface gigabitethernet0/0/2

```

Building configuration...

```

Current configuration : 104 bytes
!
interface GigabitEthernet0/0/2
 vnet trunk
 no ip address
 vnet name red
 ip bandwidth-percent eigrp 1 50
 ip hello-interval eigrp 1 5
 no ip authentication mode eigrp 1 md5
!

```

このスティッキ要素のため、**vnet name** サブモードで設定エントリを削除するには、一般にそのコマンドの **default** 形式を使用する必要があります。一部のコマンドは **no** 形式を使用しても削除できますが、これは、コマンドによって異なります。**authentication** コマンドや **summary-address** コマンドなどの一部のコマンドは **no** を使用して、コマンドを無効にします。

```

R1(config-if-vnet)# default ip authentication mode eigrp 1 md5
R1(config-if-vnet)# no ip bandwidth-percent eigrp 1
R1(config-if-vnet)# no ip hello eigrp 1
R1(config-if-vnet)# do show running-config interface g0/2

```

Building configuration...

```

Current configuration : 138 bytes
!
interface GigabitEthernet0/0/2
 vnet trunk
 no ip address
 vnet name red
!
end

```

例：マルチキャスト環境でのコマンド継承と仮想ネットワーク インターフェイス モードの上書き

次の例は、マルチキャスト ネットワークでのコマンド継承と仮想ネットワーク インターフェイス モードの上書きを示しています。トランク インターフェイスは、別々の VRF の設定要件が同じトランク インターフェイス上で類似するという事実を利用します。トランク インターフェイスに設定された適格なコマンドは、同じインターフェイス上で実行するすべての VRF に継承されます。

この例では、複数の VRF があるトランク インターフェイスに IP マルチキャスト (PIM スパース モード) が設定されています。

```
vrf definition red
  vnet tag 13
  !
  address-family ipv4
  exit-address-family
  !
ip multicast-routing
ip multicast-routing vrf red

interface GigabitEthernet0/1/0
  vnet trunk
  ip address 125.1.15.18 255.255.255.0
  ip pim sparse-mode
```

ユーザは GigabitEthernet 0/1/0 上の VRF red に IP マルチキャストを設定しないことにしたため、仮想ネットワーク インターフェイス モードの上書きを使います。IP マルチキャストは VRF red のみに対して無効にします。no ip pim コマンドは、VRF red のスパース モード、デンス モード、スパース-デンス モードを含む PIM のすべてのモードを無効にします。

```
interface GigabitEthernet0/1/0
  vnet trunk
  ip address 125.1.15.18 255.255.255.0
  ip pim sparse-mode
  vnet name red
  no ip pim
```


例 : IP マルチキャストを使用した EVN

次の例では、PIM スパース モードを設定し、RP 冗長性のため、エニーキャスト RP を利用します（この例では VRF が 1 つだけ設定されています）。

- ステップ 1** マルチキャスト ルーティングをグローバルに、かつ各 L3 インターフェイスで有効にします。黒のテキストは、グローバル テーブルを設定するコマンドのグループを示し、赤のテキストは VRF red を設定するコマンドのグループを示します。

```
ip multicast-routing
interface GigabitEthernet 1/1/1
  description GigabitEthernet to core (Global)
  ip pim sparse-mode
vrf definition red
  vnet tag 100
!
  address-family ipv4
  exit-address-family
!
ip multicast-routing vrf red
!
interface gigabitethernet1/1/1.100
  description GigabitEthernet to core (VRF red)
  vrf forwarding red
  ip pim sparse-mode
```

グローバル テーブル

VRF RED

- ステップ 2** エニーキャスト RP を使用して、VRF に RP を設定します。

```
interface loopback0
  description Anycast RP Global
  ip address 10.122.5.200 255.255.255.255
  ip pim sparse-mode
!
interface loopback1
  description MSDP Peering interface
  ip address 10.122.5.250 255.255.255.255
  ip pim sparse-mode
!
ip msdp peer 10.122.5.251 connect-source loopback 1
ip msdp originator-id loopback 1
ip pim rp-address 10.122.5.200
access-list 10 permit 239.0.0.0 0.255.255.255
!
!
interface loopback 10
  description Anycast RP VRF Red
  vrf forwarding red
  ip address 10.122.15.200 255.255.255.255
  ip pim sparse-mode
interface loopback 11
  description MSDP Peering interface VRF red
  vrf forwarding red
  ip address 10.122.15.250 255.255.255.255
  ip pim sparse-mode
!
ip msdp vrf red peer 10.122.15.251 connect-source loopback 11
ip msdp vrf red originator-id loopback 11
!
ip pim vrf red rp-address 10.122.15.200
access-list 11 permit 239.192.0.0 0.0.255.255
```

グローバル テーブル

VRF RED

その他の参考資料

関連資料

関連項目	参照先
Cisco IOS コマンド	『 Cisco IOS Master Commands List, All Releases 』
Easy Virtual Network コマンド	『 Easy Virtual Network Command Reference 』
Easy Virtual Network の概要	「 Overview of Easy Virtual Networks 」
Easy Virtual Network の共有サービスとルート レプリケーション	「 Configuring Easy Virtual Network Shared Services 」
Easy Virtual Network の管理とトラブルシューティング	「 Easy Virtual Network Management and Troubleshooting 」

MIB

MIB	MIB リンク
<p>VRF 情報を提供する MIB は引き続き Easy Virtual Network で動作します。システムのすべての VRF に関する VRF 独立 MIB のレポート情報：</p> <ul style="list-style-type: none"> • CISCO—VRF-MIB • CISCO—MVPN-MIB • MPLS-VPN MIB 	<p>選択したプラットフォーム、Cisco ソフトウェア リリース、および機能セットの MIB を検索してダウンロードする場合は、次の URL にある Cisco MIB Locator を使用します。</p> <p>http://www.cisco.com/go/mibs</p>

シスコのテクニカル サポート

説明	リンク
<p>右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。</p> <p>以下を含むさまざまな作業にこの Web サイトが役立ちます。</p> <ul style="list-style-type: none"> ・テクニカル サポートを受ける ・ソフトウェアをダウンロードする ・セキュリティの脆弱性を報告する、またはシスコ製品のセキュリティ問題に対する支援を受ける ・ツールおよびリソースへアクセスする <ul style="list-style-type: none"> - Product Alert の受信登録 - Field Notice の受信登録 - Bug Toolkit を使用した既知の問題の検索 ・Networking Professionals (NetPro) コミュニティで、技術関連のディスカッションに参加する ・トレーニング リソースへアクセスする ・TAC Case Collection ツールを使用して、ハードウェアや設定、パフォーマンスに関する一般的な問題をインタラクティブに特定および解決する <p>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。</p>	<p>http://www.cisco.com/cisco/web/support/index.html</p>

Easy Virtual Network の設定の機能情報

表 1 に、この機能のリリース履歴を示します。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェア イメージのサポート情報を検索できます。Cisco Feature Navigator を使用すると、ソフトウェア イメージがサポートする特定のソフトウェア リリース、機能セット、またはプラットフォームを確認できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。Cisco.com のアカウントは必要ありません。



(注)

表 1 には、一連のソフトウェア リリースのうち、特定の機能が初めて導入されたソフトウェア リリースだけが記載されています。特に明記していないかぎり、その機能は、一連のソフトウェア リリースの以降のリリースでもサポートされます。

表 1 Easy Virtual Network の設定の機能情報

機能名	リリース	機能情報
Easy Virtual Network トランク	Cisco IOS XE Release 3.2S	<p>このモジュールでは、仮想 IP ネットワークを設定する方法について説明します。EVN は、ネットワークのエンドツーエンドの仮想化を実現する IP ベースの仮想化テクノロジーです。単一の IP インフラストラクチャを使用して、トラフィック パスが相互に独立した状態で、個別の仮想ネットワークを提供できます。</p> <p>次のコマンドが変更されました。</p> <ul style="list-style-type: none"> • ip ospf database-filter all out • ip ospf demand-circuit • ip ospf flood-reduction • ip ospf mtu-ignore • ip ospf shutdown • ip summary-address eigrp • summary-metric • vrf definition • vrf forwarding <p>次のコマンドが導入されました。</p> <ul style="list-style-type: none"> • description (vrf 定義サブモード) • exit-if-vnet • exit-vrf-list • ip ospf vnet area • member (vrf リスト) • routing-context • show running-config vnet • show vnet • show vnet counters • show vnet tag • show vrf counters • show vrf list • vnet • vnet tag • vnet trunk • vrf list

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at www.cisco.com/go/trademarks. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1005R)

このマニュアルで使用している IP アドレスおよび電話番号は、実際のアドレスおよび電話番号を示すものではありません。マニュアル内の例、コマンド出力、ネットワーク トポロジ図、およびその他の図は、説明のみを目的として使用されています。説明の中に実際のアドレスおよび電話番号が使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

© 2010 Cisco Systems, Inc.
All rights reserved.

Copyright © 2010–2011, シスコシステムズ合同会社.
All rights reserved.



Easy Virtual Network の管理とトラブルシューティング

このモジュールでは、Easy Virtual Network (EVN) の管理とトラブルシューティング方法について説明します。

機能情報の検索

ご使用のソフトウェア リリースによっては、このモジュールに記載されている機能の中に、一部サポートされていないものがあります。最新の機能情報と注意事項については、ご使用のプラットフォームとソフトウェア リリースに対応したリリース 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、および各機能がサポートされているリリースのリストについては、「[EVN の管理とトラブルシューティングの機能情報](#)」(P.9) を参照してください。

Cisco Feature Navigator を使用すると、プラットフォーム、および Cisco ソフトウェア イメージの各サポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。Cisco.com のアカウントは必要ありません。

目次

- 「[EVN の管理とトラブルシューティングの前提条件](#)」(P.1)
- 「[EVN の管理とトラブルシューティングの概要](#)」(P.2)
- 「[EVN の管理とトラブルシューティングの方法](#)」(P.3)
- 「[参考資料](#)」(P.7)
- 「[EVN の管理とトラブルシューティングの機能情報](#)」(P.9)

EVN の管理とトラブルシューティングの前提条件

- 「[Overview of Easy Virtual Network](#)」と「[Configuring Easy Virtual Network](#)」を読み、EVN を実装します。

EVN の管理とトラブルシューティングの概要

次の概念では、EVN が VRF-Lite の機能に加えて提供する拡張機能について説明します。

- 「EXEC モードのルーティング コンテキストにより VRF の指定の繰り返しを減らす」 (P.2)
- 「traceroute コマンドの出力は VRF 名と VRF タグを示す」 (P.2)
- 「VRF 単位のデバッグ出力のフィルタリング」 (P.3)
- 「CISCO-VRF-MIB」 (P.3)

EXEC モードのルーティング コンテキストにより VRF の指定の繰り返しを減らす

複数の EXEC コマンドを発行して、単一の仮想ネットワークに適用したい場合があります。複数の EXEC コマンドに仮想ルーティングおよびフォワーディング (VRF) 名を入力する回数を減らすため、**routing-context vrf** コマンドを使用して、それらの EXEC コマンドの VRF コンテキストを 1 回設定してから、EXEC コマンドに進みます。

表 1 は、ルーティング コンテキストの有無で、Cisco IOS XE ソフトウェアの 4 つの EXEC コマンドを対比させています。左の列では、各 EXEC コマンドで VRF を指定する必要があることに注意してください。右の列では、VRF コンテキストが 1 回で指定され、プロンプトがその VRF を反映して変更されるため、コマンドごとに VRF を指定する必要はありません。

表 1 ルーティング コンテキストでの EXEC コマンド

ルーティング コンテキストのない Cisco IOS XE CLI	ルーティング コンテキスト
—	Router# routing-context vrf red Router%red#
Router# show ip route vrf red [VRF red のルーティング テーブル出力]	Router%red# show ip route [VRF red のルーティング テーブル出力]
Router# ping vrf red 10.1.1.1 [VRF red を使用した ping 結果]	Router%red# ping 10.1.1.1 [VRF red を使用した ping 結果]
Router# telnet 10.1.1.1 /vrf red [VRF red の 10.1.1.1 への Telnet]	Router%red# telnet 10.1.1.1 [VRF red の 10.1.1.1 への Telnet]
Router# traceroute vrf red 10.1.1.1 [VRF red の Traceroute 出力]	Router%red# traceroute 10.1.1.1 [VRF red の Traceroute 出力]

traceroute コマンドの出力は VRF 名と VRF タグを示す

次の図に示すように、**traceroute** コマンドの出力が、着信 VRF 名/タグおよび発信 VRF 名/タグを表示することによって、トラブルシューティングが簡単になるように拡張されました。

```
Router# traceroute vrf red 10.0.10.12
```

```
Type escape sequence to abort.  
Tracing the route to 10.0.10.12
```



```
VRF info: (vrf in name/id, vrf out name/id)
 1 10.1.13.15 (red/13,red/13) 0 msec
   10.1.16.16 (red/13,red/13) 0 msec
   10.1.13.15 (red/13,red/13) 1 msec
 2 10.1.8.13 (red/13,red/13) 0 msec
   10.1.7.13 (red/13,red/13) 0 msec
   10.1.8.13 (red/13,red/13) 0 msec
 3 10.1.2.11 (red/13,blue/10) 1 msec 0 msec 0 msec
 4 * * *
```

VRF 単位のデバッグ出力のフィルタリング

EVN を使用すると、**debug condition vrf** コマンドによって VRF 単位でデバッグ出力をフィルタできます。次に、**debug condition vrf** コマンドの出力例を示します。

```
Router# debug condition vrf red
Condition 1 set
CEF filter table debugging is on
CEF filter table debugging is on
R1#
*Aug 19 23:06:38.178: vrfmgr(0) Debug: Condition 1, vrf red triggered, count 1
R1#
```



(注) **debug condition vrf** コマンドは EIGRP 環境では機能しません。

CISCO-VRF-MIB

EVN は VRF の検出と管理のための CISCO-VRF-MIB を提供します。

EVN の管理とトラブルシューティングの方法

- 「特定の VRF への EXEC モードのルーティング コンテキストの設定」(P.3) (任意)
- 「VRF のデバッグ出力のイネーブル化」(P.5) (任意)
- 「仮想ネットワークの SNMP v2c コンテキストの設定」(P.5) (任意)
- 「仮想ネットワークへ SNMP v3 コンテキストの設定」(P.6) (任意)

特定の VRF への EXEC モードのルーティング コンテキストの設定

ルータに EXEC コマンドを発行する場合に、VRF 名の入力の繰り返しを減らすため、それらの EXEC コマンドのルーティング コンテキストを 1 回設定してから、任意の順番で EXEC コマンドに進みます。このタスクを実行して、EXEC モードのルーティング コンテキストを特定の VRF に設定し、次に、システムをグローバル EXEC コンテキストに戻します。

手順の概要

1. **enable**
2. **routing-context vrf vrf-name**
3. **show ip route [ip-address [mask] [longer-prefixes] | protocol [process-id] | static download]**

4. `ping` *[[protocol [tag] {host-name | system-address}]*
5. `telnet` *host [port]*
6. `traceroute` *[vrf vrf-name | topology topology-name] [protocol] destination*
7. `routing-context vrf global`

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<code>enable</code> 例： Router> enable	特権 EXEC モードをイネーブルにします。 • 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	<code>routing-context vrf vrf-name</code> 例： Router# routing-context vrf red	EXEC モードのルーティング コンテキストを指定した VRF にします。
ステップ 3	<code>show ip route</code> <i>[ip-address [mask] [longer-prefixes] protocol [process-id] static download]</i> 例： Router%red# show ip route	(任意) ルーティング テーブルの現在の状態を表示します。 • システム プロンプトがターゲットの VRF を反映して変更されます。 • 次の例に、vNET red のコンテキスト内で発行された show ip route コマンドを示します。vNET red のルーティング テーブルが表示されます。
ステップ 4	<code>ping</code> <i>[[protocol [tag] {host-name system-address}]</i> 例： Router%red# ping 10.1.1.1	(任意) エコー要求パケットをアドレスに送信します。 • この例では、vNET red のコンテキスト内で発行された ping コマンドを示しています。vNET red を使用した ping 結果が表示されます。
ステップ 5	<code>telnet host [port]</code> 例： Router%red# telnet 10.1.1.1	(任意) Telnet をサポートするホストにログインします。
ステップ 6	<code>traceroute</code> <i>[vrf vrf-name topology topology-name] [protocol] destination</i> 例： Router%red# traceroute 10.1.1.1	(任意) パケットを宛先に伝送するルートを表示します。
ステップ 7	<code>routing-context vrf global</code> 例： Router%red# routing-context vrf global Router>	(任意) システムをグローバル EXEC コンテキストに戻します。 • プロンプトがユーザ EXEC プロンプトに戻ります。

VRF のデバッグ出力のイネーブル化

VRF のデバッグ出力をイネーブルにするには、次の作業を実行します。

手順の概要

1. `enable`
2. `debug vrf {create | delete | error | ha | initialization | interface | ipv4 | ipv6 | issu | lock | lookup | mpls | selection}`

手順の詳細

	コマンドまたはアクション	目的
ステップ1	<code>enable</code> 例： Router> enable	特権 EXEC モードをイネーブルにします。 • 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	<code>debug vrf {create delete error ha initialization interface ipv4 ipv6 issu lock lookup mpls selection}</code> 例： Router# debug vrf ipv4	VRF デバッグ情報を表示します。

仮想ネットワークの SNMP v2c コンテキストの設定

このタスクを実行して、SNMP v2c コンテキストを VRF にマッピングします。システムによって、次の SNMP v2c 設定が自動的に実行されます。

- `snmp context` コマンドに入力した *context-name* と同じ名前を使用したコンテキスト作成 (`snmp-server context` コマンドの代わり)
- `snmp context` コマンドに入力した *community-name* と同じ名前を使用したグループ作成 (`snmp-server group` コマンドの代わり)
- `snmp context` コマンドに入力した *community-name* と同じ名前を使用したコミュニティ作成 (`snmp-server community` コマンドの代わり) デフォルトの権限は `ro` (読み取り専用) です。
- コミュニティ コンテキスト マッピング (`snmp mib community-map` コマンドの代わり)

手順の概要

1. `enable`
2. `configure terminal`
3. `ip vrf vrf-name`
4. `snmp context context-name [community community-name [rw | ro]`

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Router> enable	特権 EXEC モードをイネーブルにします。 • 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： Router# 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	ip vrf vrf-name 例： Router(config)# ip vrf green	仮想ルーティングおよびフォワーディング (VRF) インスタンスを定義し、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	snmp context context-name [community community-name [rw ro] 例： Router(config-vrf)# snmp context xxx community YYY	VRF に SNMP v2c コンテキストを設定します。 • デフォルトは ro (読み取り専用) です。

仮想ネットワークへ SNMP v3 コンテキストの設定

このタスクを実行して、SNMP v3 コンテキストを VRF にマッピングします。システムによって、次の SNMP v3 設定が自動的に実行されます。

- **snmp context** コマンドに入力した *context-name* と同じ名前を使用したコンテキスト作成 (**snmp-server context** コマンドの代わり)
- グループ作成 (**snmp-server group** コマンドの代わり) **snmp context** コマンドに入力された *context-name* に「_acnf」を付けてグループ名が生成されます。
- ユーザ作成 (**snmp-server user** コマンドの代わり) **snmp context** コマンドで設定された詳細を使用してユーザが作成されます。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip vrf vrf-name**
4. **snmp context context-name [user username [credential | [encrypted] [auth {md5 password | sha password}]] [access {access-list-number | access-list-name | ipv6 access-list-name}]]]**

手順の詳細

	コマンドまたはアクション	目的
ステップ1	<code>enable</code> 例： Router> enable	特権 EXEC モードをイネーブルにします。 • 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	<code>configure terminal</code> 例： Router# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ3	<code>ip vrf vrf-name</code> 例： Router(config)# ip vrf green	仮想ルーティングおよびフォワーディング (VRF) インスタンスを定義し、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	<code>snmp context context-name [user username [credential [encrypted] [auth {md5 password sha password}]] [access {access-list-number access-list-name ipv6 access-list-name}]</code> 例： Router(config-vrf)# snmp context green_ctx user green_comm encrypted	VRF に SNMP v3 コンテキストを設定します。

参考資料

関連資料

関連項目	参照先
Cisco IOS コマンド	『Cisco IOS Master Commands List, All Releases』
Easy Virtual Network コマンド	『Easy Virtual Network Command Reference』
Easy Virtual Network の概要	「Overview of Easy Virtual Network」
Easy Virtual Network の設定	「Configuring Easy Virtual Network」
Easy Virtual Network の共有サービスとルート レプリケーション	「Easy Virtual Network Shared Services」

MIB

MIB	MIB リンク
<p>VRF 情報を提供する MIB は引き続き Easy Virtual Network で動作します。システムのすべての VRF に関する VRF 独立 MIB のレポート情報：</p> <ul style="list-style-type: none"> • CISCO—VRF-MIB • CISCO—MVPN-MIB • MPLS-VPN MIB 	<p>選択したプラットフォーム、Cisco ソフトウェア リリース、および機能セットの MIB を検索してダウンロードする場合は、次の URL にある Cisco MIB Locator を使用します。</p> <p>http://www.cisco.com/go/mibs</p>

シスコのテクニカル サポート

説明	リンク
<p>右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。</p> <p>以下を含むさまざまな作業にこの Web サイトが役立ちます。</p> <ul style="list-style-type: none"> • テクニカル サポートを受ける • ソフトウェアをダウンロードする • セキュリティの脆弱性を報告する、またはシスコ製品のセキュリティ問題に対する支援を受ける • ツールおよびリソースへアクセスする <ul style="list-style-type: none"> - Product Alert の受信登録 - Field Notice の受信登録 - Bug Toolkit を使用した既知の問題の検索 • Networking Professionals (NetPro) コミュニティで、技術関連のディスカッションに参加する • トレーニング リソースへアクセスする • TAC Case Collection ツールを使用して、ハードウェアや設定、パフォーマンスに関する一般的な問題をインタラクティブに特定および解決する <p>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。</p>	<p>http://www.cisco.com/cisco/web/support/index.html</p>

EVN の管理とトラブルシューティングの機能情報

表 2 に、この機能のリリース履歴を示します。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェア イメージのサポート情報を検索できます。Cisco Feature Navigator を使用すると、ソフトウェア イメージがサポートする特定のソフトウェア リリース、機能セット、またはプラットフォームを確認できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。Cisco.com のアカウントは必要ありません。



(注)

表 2 には、一連のソフトウェア リリースのうち、特定の機能が初めて導入されたソフトウェア リリースだけが記載されています。特に明記していないかぎり、その機能は、一連のソフトウェア リリースの以降のリリースでもサポートされます。

表 2 EVN の管理とトラブルシューティングの機能情報

機能名	リリース	機能情報
VRF 名による VRF-Aware traceroute	Cisco IOS XE Release 3.2S	traceroute コマンドの出力が、VRF 名およびタグを表示するように拡張されました。この機能は次のセクションに記載されています。 <ul style="list-style-type: none"> 「traceroute コマンドの出力は VRF 名と VRF タグを示す」(P.2) 次のコマンドが変更されました： traceroute
Easy Virtual Network トランク	Cisco IOS XE Release 3.2S	ユーザは debug condition vrf コマンドを使用して、VRF 単位でデバッグ出力をフィルタできます。この機能は次のセクションに記載されています。 <ul style="list-style-type: none"> 「VRF 単位のデバッグ出力のフィルタリング」(P.3) 次のコマンドが導入されました。 <ul style="list-style-type: none"> debug condition vrf debug vrf
Easy Virtual Network MIB およびコンテキストベースの SNMP の簡素化	Cisco IOS XE Release 3.2S	EVN MIB およびコンテキストベースの SNMP により SNMP の設定が簡単になります。この機能は次のセクションに記載されています。 <ul style="list-style-type: none"> 「仮想ネットワークの SNMP v2c コンテキストの設定」(P.5) 「仮想ネットワークへ SNMP v3 コンテキストの設定」(P.6) 次のコマンドが変更されました： snmp context

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at www.cisco.com/go/trademarks. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1005R)

このマニュアルで使用している IP アドレスおよび電話番号は、実際のアドレスおよび電話番号を示すものではありません。マニュアル内の例、コマンド出力、ネットワーク トポロジ図、およびその他の図は、説明のみを目的として使用されています。説明の中に実際のアドレスおよび電話番号が使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

© 2010 Cisco Systems, Inc.
All rights reserved.

Copyright © 2010–2011, シスコシステムズ合同会社.
All rights reserved.



Easy Virtual Network 共有サービスの設定

このモジュールでは、Easy Virtual Network (EVN) でルート レプリケーションとルート再配布を使用してサービスを共有する方法について説明します。

機能情報の検索

ご使用のソフトウェア リリースによっては、このモジュールに記載されている機能の中に、一部サポートされていないものがあります。最新の機能情報と注意事項については、ご使用のプラットフォームとソフトウェア リリースに対応したリリース 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、および各機能がサポートされているリリースのリストについては、「[Easy Virtual Network 共有サービスの機能情報](#)」(P.23) を参照してください。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェア イメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。Cisco.com のアカウントは必要ありません。

目次

- 「[仮想 IP ネットワーク共有サービスの前提条件](#)」(P.1)
- 「[仮想 IP ネットワーク共有サービスの制約事項](#)」(P.2)
- 「[Easy Virtual Network 共有サービスの概要](#)」(P.2)
- 「[Easy Virtual Network を使用してサービスを共有する方法](#)」(P.5)
- 「[EVN 共有サービスの設定例](#)」(P.15)
- 「[関連資料](#)」(P.21)
- 「[Easy Virtual Network 共有サービスの機能情報](#)」(P.23)

仮想 IP ネットワーク共有サービスの前提条件

- 「[Overview of Easy Virtual Networks](#)」を参照してください。
- 「[Configuring Easy Virtual Networks](#)」に基づいて、Easy Virtual Network を実装します。

仮想 IP ネットワーク共有サービスの制約事項

ルート レプリケーションは、スタティック ルート、EIGRP ルート、および OSPF ルートでサポートされています。BGP 間でルートをレプリケートすることは不可能ですが、VRF 間でルートをコピーする BGP のインポートおよびエクスポート方法が仮想ネットワークで使用できるため、問題になりません。

Easy Virtual Network 共有サービスの概要

- 「Easy Virtual Network の共有サービス」(P.2)
- 「Easy Virtual Network は VRF-Lite よりも共有サービスを簡単にする」(P.2)
- 「EVN でのルート レプリケーション プロセス」(P.3)
- 「EVN のルート レプリケーションの動作」(P.4)
- 「EVN でのルート レプリケーション後のルート プリファレンス ルール」(P.5)

Easy Virtual Network の共有サービス

複数の仮想ネットワークがアクセスする必要がある共通のサービス（データベース サーバやアプリケーション サーバなど）がいくつかあります。共有サービスは次の理由のため、利点があります。

- 通常、グループごとにサービスが重複しません。
- 共有サービスは経済的です。
- 共有サービスは効率的で管理が容易です。
- ポリシーを中央で展開できます。

ルートの分割を実現するには、物理的または仮想的に、各仮想ネットワークに 1 つずつサービスを複製することができます。ただし、そのソリューションは、費用効果が高くないか、実現不可能である場合があります。EVN をサポートするルータの場合、ソリューションはルート レプリケーションとルート再配布を実行することです。

ルート レプリケーションにより、ルートが仮想ネットワーク間でレプリケートされ、ある仮想ネットワークに存在するクライアントが、別の仮想ネットワークに存在するプレフィックスに到達できるため、共有サービスが可能になります。

共有サービス アプローチは、DNS、DHCP、および企業の通信に最適です。インターネット ゲートウェイへの共有アクセス用のソリューションではありません。

Easy Virtual Network は VRF-Lite よりも共有サービスを簡単にする

VRF-Lite の共有サービスでは、ルート識別子 (RD)、インポートおよびエクスポートによるルートターゲット、および BGP の設定が必要です。

EVN 環境では、簡単な展開であるルート レプリケーションによって、ルート共有サービスを実現できます。ルート レプリケーションでは BGP、RD、ルート ターゲット、インポートやエクスポートも必要ありません。

要約すると、VRF 間でルートをコピーするための BGP インポートおよびエクスポート方法は VRF-Lite と EVN の両方で機能します。ただし、ルート レプリケーションは、複数の仮想ネットワークで共通サービスの共有を可能にするより簡単な代替方法です。

EVN でのルート レプリケーション プロセス

共有サービスでは、クライアントとサーバが異なる仮想ネットワークに存在します。クライアントとサーバ間の接続を実現するには、仮想ネットワーク間でルートが交換される必要があります。VRF 間のそのようなルート交換は、VRF-Lite が実装されているか EVN が実装されているかに応じて、2 つの方法で実現できます。

- VRF-Lite では、ルート インポート/エクスポート機能を使用して、BGP によってルート リークが実現されます。
- EVN では、ルート レプリケーションが RIB によって直接サポートされるため、BGP への依存関係はありません。ルートが異なる仮想ネットワークからレプリケートされると、Interior Gateway Protocol (IGP) への既存の再配布により、それらのルートが各仮想ネットワーク全体に伝播されます。

次のルート レプリケーション シナリオでは、ルータに Services と User-A という名前の 2 つの VRF があります。OSPF が次のように設定されます。

```
router ospf 99 vrf services
 network 126.1.0.0 0.0.255.255 area 0
!
router ospf 98 vrf user-a
 network 126.1.0.0 0.0.255.255 area 0
```

さらに、VRF User-A にルート再配布が設定されます。

```
vrf definition user-a
!
 address-family ipv4
 route-replicate from vrf services unicast ospf 99
 exit-address-family
```

このシナリオでは、VRF サービスの次の RIB に 4 つのルートが含まれ、それらのうち 3 つは VRF User-A の RIB にレプリケートされます。ルート レプリケーションでは、[図 1](#) に示すように、ソース RIB へのリンクが作成されます。

図 1 ルート レプリケーション

RIB—VRF Services

ルート	タイプ	宛先 インターフェイス	ネクスト ホップ
126.1.17.0/24	Connected	Gi0/1	
126.1.9.0/24	OSPF	Gi0/1	126.1.17.13
126.1.12.0/24	OSPF	Gi0/1	126.1.17.13
126.1.14.0/24	OSPF	Gi0/1	126.1.17.13

RIB—VRF User-A

ルート	タイプ	宛先 インターフェイス	ネクスト ホップ
126.1.9.0/24	OSPF	Gi0/1	126.1.17.13
126.1.12.0/24	OSPF	Gi0/1	126.1.17.13
126.1.14.0/24	OSPF	Gi0/1	126.1.17.13

ルート レプリケーションの設定により、仮想 IP ネットワーク間の相互の再配布が可能になります。共有サービスの場合、共有サービスにアクセスする必要がある VRF 内にルート レプリケーションを設定することがあります。各 **route-replicate** コマンド内で、ルーティング ループを避けるため、ルート マップを使用して、任意でルートをフィルタで除外できます。つまり、ルートを元のルーティング プロトコルに再配布しないようにできます。ネィティブ ルートをレプリケートされたルートとして表示しないようにできます。

ルート レプリケーションを実装する場所

ルート レプリケーションは、できるだけ共有サービスの近くのルータに実装することをお勧めします。理想的には、サーバ サブネットに直接接続されているルータを使用して、サーバ VRF 上のホスト プレフィックスを再配布する必要をなくし、それにより、ルーティング ループの可能性を回避する必要があります。

EVN のルート レプリケーションの動作

このセクションでは、EVN のルート レプリケーションの動作について説明します。この動作は Multi-Topology ルーティングの動作と異なります。EVN 環境では、

- **route-replicate** コマンドは **address-family ipv4** コマンドでのみ受け入れられ、**vrf definition** コマンドで設定されます。
- **route-replicate** コマンドは、指定されたアドレス ファミリ内のベース トポロジにルートをレプリケートします。
- ソース プロトコルとして **all** を指定した場合、指定した宛先トポロジに対して、VRF あたり 1 つの **route-replicate** コマンドのみが許可されます。
- **no route-replicate** コマンドは、ソース プロトコルを除外できます。

- ソース プロトコルとして **all** を指定した場合、接続されているルートがレプリケートされます (**route-replicate** コマンドの Multi-Topology ルーティング バージョンと異なる)。
- レプリケートされたルートは、ソース ルートのアドミニストレーティブ ディスタンスとソース プロトコルを継承します。

EVN でのルート レプリケーション後のルート プリファレンス ルール

ルートがレプリケートされる場合、次のルールによってルート プリファレンスが決定します。

1. 2 つのルートが同じプロトコルで所有されていて (たとえば、両方とも OSPF ルートである)、両方のルートのソース VRF が同じであり、一方のルートがレプリケートされていない場合、レプリケートされていないルートが優先されます。

上のルールが適用しない場合、次のルールによって、この順番でルート プリファレンスが決定されます。

2. アドミニストレーティブ ディスタンスが小さいルートが優先されます。
3. デフォルトのアドミニストレーティブ ディスタンスが小さいルートが優先されます。
4. レプリケートされたルートよりもレプリケートされていないルートが優先されます。
5. 元の vrf 名を比較します。辞書編集上で vrf 名が小さいルートが優先されます。
6. 元のサブアドレスファミリを比較します。マルチキャストよりユニキャストが優先されます。
7. 最も古いルートが優先されます。

Easy Virtual Network を使用してサービスを共有する方法

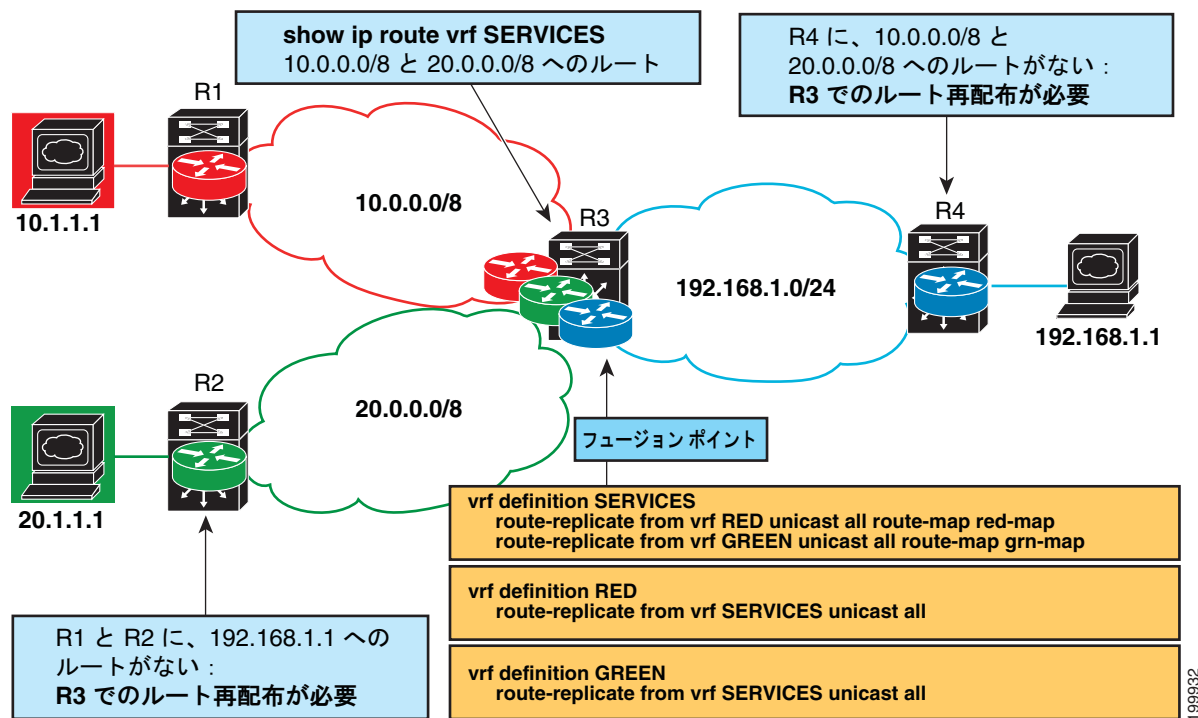
2 つの VRF から共有サービス VRF への完全なアクセス可能性を実現するには、次の両方のタスクを実行します。

- 「EVN でサービスを共有するためのルート レプリケーションの設定」(P.5) (必須)
- 「EVN の共有サービスへの再配布の設定」(P.12) (必須)

EVN でサービスを共有するためのルート レプリケーションの設定

次のタスクを実行して、ある VRF から別の VRF にルートをレプリケートします。このタスク テーブル内の例は図 2 に基づいています。

図 2 ルートレプリケーションを使用した共有サービス



この特定のタスクでは、VRF SERVICES からのルートが VRF RED にレプリケートされます。双方向トラフィックを許可するため、VRF RED からのルートも VRF SERVICES にレプリケートされます。



(注)

実際の EVN 環境では、VRF SERVICES と 3 つ目の VRF 間、およびもっと多くの VRF 間のルートレプリケーションが存在することがあります。そうしたレプリケーションは、簡潔さのため、次の設定タスクから外しています。

手順の概要

1. enable
2. configure terminal
3. vrf definition *vrf-name*
4. vnet tag *number*
5. description *string*
6. address-family ipv4
7. exit
8. exit
9. vrf definition *vrf-name*
10. vnet tag *number*
11. description *string*
12. address-family ipv4
13. exit

14. **exit**
15. **interface** *type number*
16. **vrf forwarding** *vrf-name*
17. **ip address** *ip-address mask*
18. **no shutdown**
19. **exit**
20. **router ospf** *process-id vrf vrf-name*
21. **network** *ip-address wildcard-mask area area-id*
22. **exit**
23. **router ospf** *process-id vrf vrf-name*
24. **network** *ip-address wildcard-mask area area-id*
25. **exit**
26. **vrf definition** *vrf-name*
27. **address-family** *ipv4*
28. **route-replicate from** [*vrf vrf-name*] {**multicast** | **unicast**} {**all** | *protocol-name*} [**route-map** *map-tag*]
29. **exit**
30. **exit**
31. **vrf definition** *vrf-name*
32. **address-family** *ipv4*
33. **route-replicate from** [*vrf vrf-name*] {**multicast** | **unicast**} {**all** | *protocol-name*} [**route-map** *map-tag*]
34. **end**
35. **show ip route vrf** *vrf-name*

手順の詳細

	コマンドまたはアクション	目的
ステップ1	enable 例： Router> enable	特権 EXEC モードをイネーブルにします。 • 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： Router# 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	vrf definition <i>vrf-name</i> 例： Router(config)# vrf definition SERVICES	VRF を定義します。

Easy Virtual Network を使用してサービスを共有する方法

	コマンドまたはアクション	目的
ステップ 4	<code>vnet tag number</code> 例： Router(config-vrf)# vnet tag 100	VRF のグローバルな数値タグを指定します。 • 同じタグ番号を各エッジおよびトランク インターフェイス上の同じ VRF に設定する必要があります。
ステップ 5	<code>description string</code> 例： Router(config-vrf)# description shared services	(任意) コンフィギュレーション ファイルを参照するネットワーク管理者に役立つように、VRF について記述します。
ステップ 6	<code>address-family ipv4</code> 例： Router(config-vrf)# address-family ipv4	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 アドレス プレフィ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 7	<code>exit</code> 例： Router(config-vrf-af)# exit	次に高いコンフィギュレーション モードに移動します。
ステップ 8	<code>exit</code> 例： Router(config-vrf)# exit	次に高いコンフィギュレーション モードに移動します。
ステップ 9	<code>vrf definition vrf-name</code> 例： Router(config)# vrf definition RED	VRF を定義します。
ステップ 10	<code>vnet tag number</code> 例： Router(config-vrf)# vnet tag 200	VRF のグローバルな数値タグを指定します。 • 同じタグ番号を各エッジおよびトランク インターフェイス上の同じ VRF に設定する必要があります。
ステップ 11	<code>description string</code> 例： Router(config-vrf)# description user of services	(任意) コンフィギュレーション ファイルを参照するネットワーク管理者に役立つように、VRF について記述します。
ステップ 12	<code>address-family ipv4</code> 例： Router(config-vrf)# address-family ipv4	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 アドレス プレフィ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 13	<code>exit</code> 例： Router(config-vrf-af)# exit	次に最高のコンフィギュレーション モードに移動します。

	コマンドまたはアクション	目的
ステップ 14	exit 例： Router(config-vrf)# exit	次に最高のコンフィギュレーション モードに移動します。
ステップ 15	interface type number 例： Router(config)# interface gigabitethernet 0/0/0	インターフェイス タイプと番号を設定します。
ステップ 16	vrf forwarding vrf-name 例： Router(config-if)# vrf forwarding SERVICES	インターフェイスと VRF インスタンスを関連付けます。
ステップ 17	ip address ip-address mask 例： Router(config-if)# ip address 192.168.1.3 255.255.255.0	インターフェイスに対するプライマリ IP アドレスを設定します。
ステップ 18	no shutdown 例： Router(config-if)# no shutdown	インターフェイスを再起動します。
ステップ 19	exit 例： Router(config-vrf)# exit	次に最高のコンフィギュレーション モードに移動します。
ステップ 20	router ospf process-id vrf vrf-name 例： Router(config)# router ospf 99 vrf SERVICES	OSPF ルーティング プロセスを設定します。 <ul style="list-style-type: none"> この例では OSPF を使用していますが、EIGRP も使用できます。
ステップ 21	network ip-address wildcard-mask area area-id 例： Router(config-router)# network 192.168.1.0 0.0.0.255 area 0	OSPF が実行するインターフェイスと、それらのインターフェイスに対するエリア ID を定義します。
ステップ 22	exit 例： Router(config-router)# exit	次に高いコンフィギュレーション モードに移動します。
ステップ 23	router ospf process-id [vrf vrf-name] 例： Router(config)# router ospf 98 vrf RED	OSPF ルーティング プロセスを設定します。

Easy Virtual Network を使用してサービスを共有する方法

	コマンドまたはアクション	目的
ステップ 24	<pre>network ip-address wildcard-mask area area-id</pre> <p>例： Router(config-router)# network 192.168.1.0 0.0.0.255 area 0</p>	OSPF が実行するインターフェイスと、それらのインターフェイスに対するエリア ID を定義します。
ステップ 25	<pre>exit</pre> <p>例： Router(config-router)# exit</p>	次に高いコンフィギュレーション モードに移動します。
ステップ 26	<pre>vrf definition vrf-name</pre> <p>例： Router(config)# vrf definition RED</p>	VRF を定義します。
ステップ 27	<pre>address-family ipv4</pre> <p>例： Router(config-vrf)# address-family ipv4</p>	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 アドレス プレフィ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 28	<pre>route-replicate from [vrf vrf-name] {multicast unicast} {all protocol-name} [route-map map-tag]</pre> <p>例： Router(config-vrf-af)# route replicate from vrf SERVICES unicast all</p>	<p>指定したアドレス ファミリ内のベース トポロジにルートをレプリケートします。</p> <ul style="list-style-type: none"> ソース プロトコルとして all キーワードを指定した場合、指定した宛先トポロジに対して、VRF あたり 1 つの route-replicate コマンドのみが許可されます。 接続されたルートのみをレプリケートするには、ソース <i>protocol-name</i> として connected キーワードを使用します。
ステップ 29	<pre>exit</pre> <p>例： Router(config-vrf-af)# exit</p>	次に高いコンフィギュレーション モードに移動します。
ステップ 30	<pre>exit</pre> <p>例： Router(config-vrf)# exit</p>	次に高いコンフィギュレーション モードに移動します。
ステップ 31	<pre>vrf definition vrf-name</pre> <p>例： Router(config)# vrf definition SERVICES</p>	VRF を定義します。

	コマンドまたはアクション	目的
ステップ 32	<code>address-family ipv4</code> 例： Router(config-vrf)# address-family ipv4	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 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 33	<code>route-replicate from [vrf vrf-name] {multicast unicast} {all protocol-name} [route-map map-tag]</code> 例： Router(config-vrf-af)# route replicate from vrf RED unicast all	指定したアドレス ファミリ内のベース トポロジにルートをレプリケートします。 • これは、双方向トラフィックを許可するステップ 28 と逆のレプリケーションです。
ステップ 34	<code>end</code> 例： Router(config-vrf-af)# end	コンフィギュレーション モードを終了します。
ステップ 35	<code>show ip route vrf vrf-name</code> 例： Router# show ip route vrf RED	(任意) プラス記号 (+) で示されているレプリケートされたルートを含めて、ルートを表示します。

例

次に、前のタスク テーブルのタスクに基づいた `show ip route vrf` コマンドの出力を示します。

```
Router# show ip route vrf RED
```

```
Routing Table: RED
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override
```

```
Gateway of last resort is not set
```

```
          192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C   +   192.168.1.0/24 is directly connected (SERVICES), GigabitEthernet0/0/0
L   +   192.168.1.3/32 is directly connected (SERVICES), GigabitEthernet0/0/0
Router#
```

この次の手順

タスク「EVN でサービスを共有するためのルート レプリケーションの設定」(および図 2 の VRF GREEN のパラレル タスク) の実行後、Router 3 には 10.0.0.0/8 と 20.0.0.0/8 へのルートが設定されます。さらに、Router 1 と Router 2 には 192.168.1.0/24 へのルートが設定されます。

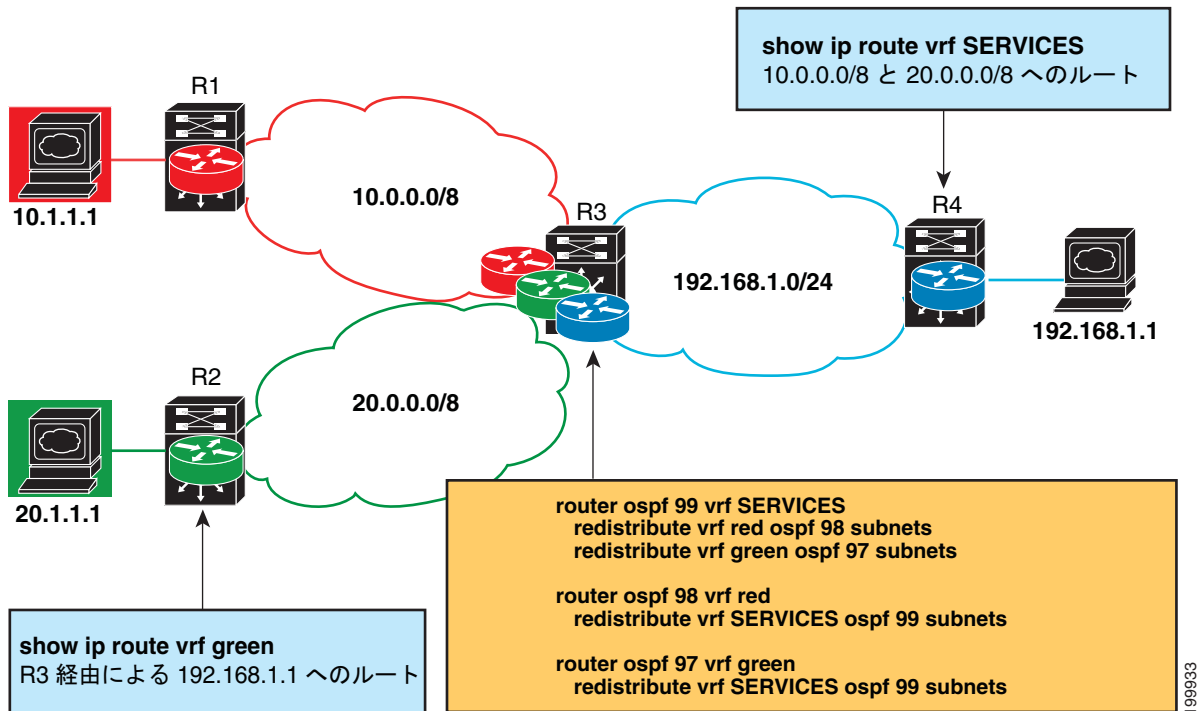
しかし、Router 1 と Router 2 にはまだ 192.168.1.1 に存在する共有サービスへのルートが設定されていません。さらに、Router 4 には 10.0.0.0/8 と 20.0.0.0/8 へのルートが設定されていません。それらのアクセスには、次のタスク「EVN の共有サービスへの再配布の設定」で実行するルート再配布が必要です。

EVN の共有サービスへの再配布の設定

このタスクは、タスク「EVN でサービスを共有するためのルート レプリケーションの設定」も実行しているという仮定に基づいています。

図 3 は図 2 で使用した同じネットワークを示しています。このタスクでは、Router 1 と Router 2 に 192.168.1.1 に存在する共有サービスへのルートが設定されるように、Router 3 で再配布を実行します。

図 3 再配布を使用した共有サービス



手順の概要

1. enable
2. configure terminal
3. router ospf *process-id* vrf *vrf-name*
4. redistribute vrf *vrf-name* ospf *process-id* subnets
5. redistribute vrf *vrf-name* ospf *process-id* subnets
6. exit
7. router ospf *process-id* vrf *vrf-name*
8. redistribute vrf *vrf-name* ospf *process-id* subnets
9. exit
10. router ospf *process-id* vrf *vrf-name*
11. redistribute vrf *vrf-name* ospf *process-id* subnets

12. end

13. show ip route vrf *vrf-name*

手順の詳細

	コマンドまたはアクション	目的
ステップ1	enable 例： Router> enable	特権 EXEC モードをイネーブルにします。 • 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： Router# 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	router ospf process-id vrf vrf-name 例： Router(config)# router ospf 99 vrf SERVICES	OSPF ルーティング プロセスを設定します。
ステップ4	redistribute vrf vrf-name ospf process-id subnets 例： Router(config-router)# redistribute vrf RED ospf 98 subnets	あるルーティング ドメインから別のルーティング ドメインにルートを再配布します。
ステップ5	redistribute vrf vrf-name ospf process-id subnets 例： Router(config-router)# redistribute vrf GREEN ospf 97 subnets	あるルーティング ドメインから別のルーティング ドメインにルートを再配布します。
ステップ6	exit 例： Router(config-router)# exit	次に最高のコンフィギュレーション モードに移動します。
ステップ7	router ospf process-id vrf vrf-name 例： Router(config)# router ospf 98 vrf RED	OSPF ルーティング プロセスを設定します。
ステップ8	redistribute vrf vrf-name ospf process-id subnets 例： Router(config-router)# redistribute vrf SERVICES ospf 99 subnets	あるルーティング ドメインから別のルーティング ドメインにルートを再配布します。

Easy Virtual Network を使用してサービスを共有する方法

	コマンドまたはアクション	目的
ステップ 9	<code>exit</code> 例： Router(config-router)# exit	次に最高のコンフィギュレーション モードに移動します。
ステップ 10	<code>router ospf process-id vrf vrf-name</code> 例： Router(config)# router ospf 97 vrf GREEN	OSPF ルーティング プロセスを設定します。
ステップ 11	<code>redistribute vrf vrf-name ospf process-id subnets</code> 例： Router(config-router)# redistribute vrf SERVICES ospf 99 subnets	あるルーティング ドメインから別のルーティング ドメインにルートを再配布します。
ステップ 12	<code>end</code> 例： Router(config-router)# end	コンフィギュレーション モードを終了します。
ステップ 13	<code>show ip route vrf vrf-name</code> 例： Router# show ip route vrf RED	(任意) プラス記号 (+) で示されているレプリケートされたルートを含めて、ルートを表示します。

EVN 共有サービスの設定例

- 「例：マルチキャスト環境での EVN ルート レプリケーションとルート再配布」(P.15)

例：マルチキャスト環境での EVN ルート レプリケーションとルート再配布

図 4 マルチキャストの物理トポロジの例

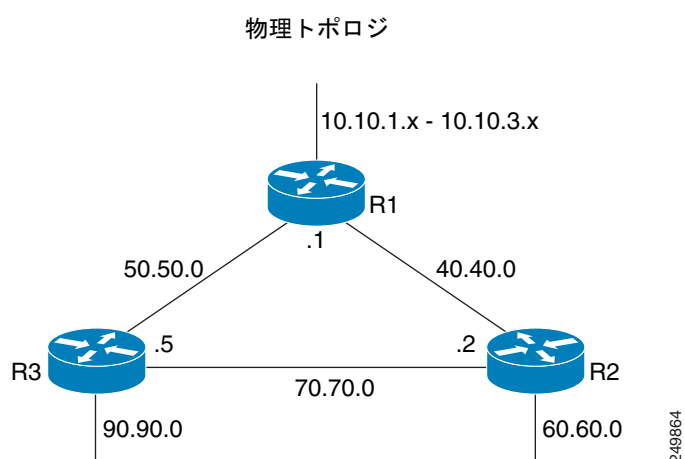


図 5 VRF Blue トポロジ
VRF blue トポロジ

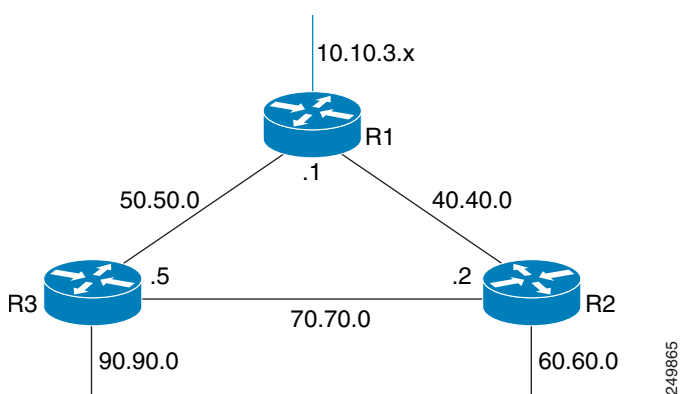
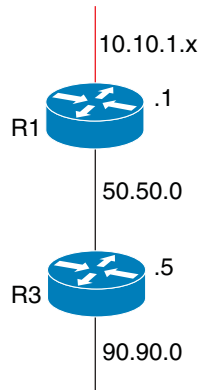


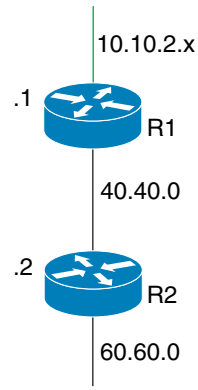
図 6 VRF Red および VRF Green トポロジ

VRF red トポロジ



vrf red に参加しているグループ：
 Sred、G1：(10.10.1.200、232.1.1.1)
 Sblue、G2：(10.10.3.202、232.3.3.3)

VRF green トポロジ



vrf green に参加しているグループ：
 Sgreen、G1：(10.10.1.201、232.1.1.1)
 Sblue、G2：(10.10.3.202、232.3.3.3)

249866

図 4、図 5、および図 6 には、3つのマルチキャストストリームがあります。

- Sred、G1：(10.10.1.200、232.1.1.1)：VRF red のソースとレシーバー
- Sgreen、G1：(10.10.2.201、232.1.1.1)：VRF green のソースとレシーバー
- Sblue、G2：(10.10.3.202、232.3.3.3)：VRF red と green の blue のソースとレシーバー

VRF blue (10.10.3.0/24) のサーバプレフィックスは、R3 と R2 の VRF にレプリケートされ、配布されます。

マルチキャストグループ 232.3.3.3 と VRF blue のそのソースは、VRF red と VRF green の両方にレシーバーがあります。ストリームは共有 VRF (blue) 経由で送信され、R3 の VRF と R2 の VRF green にレプリケートされます。

R1 の設定

```
vrf definition blue
  vnet tag 4
  !
  address-family ipv4
  exit-address-family
  !
vrf definition green
  vnet tag 3
  !
  address-family ipv4
  exit-address-family
  !
vrf definition red
  vnet tag 2
  !
  address-family ipv4
  exit-address-family
  !
vrf list vnet-list1
  member blue
  member red
  !
```



```
vrf list vnet-list2
  member blue
  member green
!
vrf list vnet-list3
  member blue
!
ip multicast-routing distributed
ip multicast-routing vrf red distributed
ip multicast-routing vrf green distributed
ip multicast-routing vrf blue distributed
!
interface FastEthernet0/0/2
  vnet trunk list vnet-list1
  ip address 50.50.0.1 255.255.255.0
  no ip redirects
  no ip proxy-arp
  ip pim sparse-dense-mode
!
interface GigabitEthernet0/1/1
  vnet trunk list vnet-list2
  ip address 40.40.0.1 255.255.255.0
  no ip redirects
  no ip proxy-arp
  ip pim sparse-dense-mode
!
interface GigabitEthernet0/1/3
  ip address 10.10.0.1 255.255.255.0
  no ip redirects
  no ip proxy-arp
  ip pim sparse-dense-mode
!
interface GigabitEthernet0/1/3.2
  vrf forwarding red
  encapsulation dot1Q 2
  ip address 10.10.1.1 255.255.255.0
  ip pim sparse-dense-mode
!
interface GigabitEthernet0/1/3.3
  vrf forwarding green
  encapsulation dot1Q 3
  ip address 10.10.2.1 255.255.255.0
  ip pim sparse-dense-mode
!
interface GigabitEthernet0/1/3.4
  vrf forwarding blue
  encapsulation dot1Q 4
  ip address 10.10.3.1 255.255.255.0
  ip pim sparse-dense-mode
!
router ospf 201 vrf red
  nsf
  redistribute connected subnets
  network 10.10.1.0 0.0.0.255 area 0
  network 50.50.0.0 0.0.0.255 area 0
!
router ospf 202 vrf green
  nsf
  network 10.10.2.0 0.0.0.255 area 0
  network 40.40.0.0 0.0.0.255 area 0
!
router ospf 203 vrf blue
  router-id 11.11.11.11
  nsf
```

[vnet trunk for red and blue]

[vnet trunk for green and blue]

```

network 10.10.3.0 0.0.0.255 area 0
network 40.40.0.0 0.0.0.255 area 0
network 50.50.0.0 0.0.0.255 area 0
!
router ospf 200
nsf
redistribute connected subnets
network 10.10.0.0 0.0.0.255 area 0
network 40.40.0.0 0.0.0.255 area 0
network 50.50.0.0 0.0.0.255 area 0
!
ip pim ssm default
ip pim vrf red ssm default
ip pim vrf green ssm default
ip pim vrf blue ssm default
!

```

R2 の設定

```

vrf definition blue
vnet tag 4
!
address-family ipv4
exit-address-family
!
vrf definition green
vnet tag 3
!
address-family ipv4
route-replicate from vrf blue unicast all route-map blue-map [replicate routes from blue
to green]
exit-address-family
!
vrf definition red
vnet tag 2
!
address-family ipv4
exit-address-family
!
vrf list vnet-list1
member blue
member green
!
vrf list vnet-list2
member blue
!
ip multicast-routing distributed
ip multicast-routing vrf red distributed
ip multicast-routing vrf green distributed
ip multicast-routing vrf blue distributed
!
interface FastEthernet0/0/6
vnet trunk list vnet-list2 [vnet trunk for blue]
ip address 70.70.0.2 255.255.255.0
no ip redirects
no ip proxy-arp
ip pim sparse-dense-mode
!
interface GigabitEthernet0/1/2
vnet trunk list vnet-list1[vnet trunk for green and blue]
ip address 40.40.0.2 255.255.255.0
no ip redirects
no ip proxy-arp
ip pim sparse-dense-mode

```

```
!  
interface GigabitEthernet0/1/4  
  vnet trunk list vnet-list1 [vnet trunk for green and blue]  
  ip address 60.60.0.2 255.255.255.0  
  no ip redirects  
  no ip proxy-arp  
  ip pim sparse-dense-mode  
!  
router ospf 202 vrf green  
  redistribute connected subnets  
  redistribute vrf blue ospf 203 subnets route-map blue-map [redistribute routes replicated  
from blue in red]  
  network 40.40.0.0 0.0.0.255 area 0  
  network 60.60.0.0 0.0.0.255 area 0  
!  
router ospf 203 vrf blue  
  router-id 22.22.22.22  
  network 40.40.0.0 0.0.0.255 area 0  
  network 60.60.0.0 0.0.0.255 area 0  
  network 70.70.0.0 0.0.0.255 area 0  
!  
router ospf 200  
  redistribute connected subnets  
  network 40.40.0.0 0.0.0.255 area 0  
  network 60.60.0.0 0.0.0.255 area 0  
  network 70.70.0.0 0.0.0.255 area 0  
!  
ip pim ssm default  
ip pim vrf red ssm default  
ip pim vrf green ssm default  
ip pim vrf blue ssm default  
!  
ip prefix-list server-prefix seq 5 permit 10.10.3.0/24  
!  
route-map blue-map permit 10  
  match ip address prefix-list server-prefix  
!
```

R3 の設定

```
vrf definition blue  
  vnet tag 4  
  !  
  address-family ipv4  
  exit-address-family  
!  
vrf definition green  
  vnet tag 3  
  !  
  address-family ipv4  
  exit-address-family  
!  
vrf definition red  
  vnet tag 2  
  !  
  address-family ipv4  
  route-replicate from vrf blue unicast all route-map blue-map [replicate routes from  
blue to red]  
  exit-address-family  
!  
vrf list vnet-list1  
  member blue  
  member red  
!
```

```

vrf list vnet-list2
  member blue
!
ip multicast-routing distributed
ip multicast-routing vrf red distributed
ip multicast-routing vrf green distributed
ip multicast-routing vrf blue distributed
!
interface GigabitEthernet0/2/0
  vnet trunk list vnet-list1                [vnet trunk for red and blue]
  ip address 90.90.0.5 255.255.255.0
  no ip redirects
  no ip proxy-arp
  ip pim sparse-dense-mode
!
interface GigabitEthernet1/2/0
  vnet trunk list vnet-list1                [vnet trunk for red and blue]
  ip address 50.50.0.5 255.255.255.0
  no ip redirects
  no ip proxy-arp
  ip pim sparse-dense-mode
!
interface FastEthernet2/0/0
  vnet trunk list vnet-list2                [vnet trunk for blue]
  ip address 70.70.0.5 255.255.255.0
  no ip redirects
  no ip proxy-arp
  ip pim sparse-dense-mode
!
router ospf 201 vrf red
  redistribute connected subnets
  redistribute vrf blue ospf 203 subnets route-map blue-map    [redistribute routes
replicated from blue in red]
  network 50.50.0.0 0.0.0.255 area 0
  network 90.90.0.0 0.0.0.255 area 0
!
router ospf 203 vrf blue
  router-id 55.55.55.55
  network 50.50.0.0 0.0.0.255 area 0
  network 70.70.0.0 0.0.0.255 area 0
  network 90.90.0.0 0.0.0.255 area 0
!
router ospf 200
  redistribute connected subnets
  network 50.50.0.0 0.0.0.255 area 0
  network 70.70.0.0 0.0.0.255 area 0
  network 90.90.0.0 0.0.0.255 area 0
!
ip pim ssm default
ip pim vrf red ssm default
ip pim vrf green ssm default
ip pim vrf blue ssm default
!
ip prefix-list server-prefix seq 5 permit 10.10.3.0/24
!
route-map blue-map permit 10
  match ip address prefix-list server-prefix
!

```

参考資料

関連資料

関連項目	参照先
Cisco IOS コマンド	『 Cisco IOS Master Commands List, All Releases 』
Easy Virtual Network コマンド	『 Easy Virtual Network Command Reference 』
Easy Virtual Network の概要	「 Overview of Easy Virtual Network 」
Easy Virtual Network の設定	「 Configuring Easy Virtual Network 」
Easy Virtual Network の管理とトラブルシューティング	「 Easy Virtual Network Management and Troubleshooting 」

MIB

MIB	MIB リンク
<p>VRF 情報を提供する MIB は引き続き Easy Virtual Network で動作します。システムのすべての VRF に関する VRF 独立 MIB のレポート情報：</p> <ul style="list-style-type: none"> • CISCO—VRF-MIB • CISCO—MVPN-MIB • MPLS-VPN MIB 	<p>選択したプラットフォーム、Cisco ソフトウェア リリース、および機能セットの MIB を検索してダウンロードする場合は、次の URL にある Cisco MIB Locator を使用します。</p> <p>http://www.cisco.com/go/mibs</p>

シスコのテクニカル サポート

説明	リンク
<p>右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。</p> <p>以下を含むさまざまな作業にこの Web サイトが役立ちます。</p> <ul style="list-style-type: none">・テクニカル サポートを受ける・ソフトウェアをダウンロードする・セキュリティの脆弱性を報告する、またはシスコ製品のセキュリティ問題に対する支援を受ける・ツールおよびリソースへアクセスする<ul style="list-style-type: none">- Product Alert の受信登録- Field Notice の受信登録- Bug Toolkit を使用した既知の問題の検索・Networking Professionals (NetPro) コミュニティで、技術関連のディスカッションに参加する・トレーニング リソースへアクセスする・TAC Case Collection ツールを使用して、ハードウェアや設定、パフォーマンスに関する一般的な問題をインタラクティブに特定および解決する <p>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。</p>	<p>http://www.cisco.com/cisco/web/support/index.html</p>

Easy Virtual Network 共有サービスの機能情報

表 1 に、この機能のリリース履歴を示します。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェア イメージのサポート情報を検索できます。Cisco Feature Navigator を使用すると、ソフトウェア イメージがサポートする特定のソフトウェア リリース、機能セット、またはプラットフォームを確認できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。Cisco.com のアカウントは必要ありません。



(注)

表 1 には、一連のソフトウェア リリースのうち、特定の機能が初めて導入されたソフトウェア リリースだけが記載されています。特に明記していないかぎり、その機能は、一連のソフトウェア リリースの以降のリリースでもサポートされます。

表 1 Easy Virtual Network 共有サービスの機能情報

機能名	リリース	機能情報
VRF ルート レプリケーション	Cisco IOS XE Release 3.2S	<p>このモジュールでは、Easy Virtual Network 環境でルート レプリケーションと再配布を使用してサービスを共有する方法について説明します。</p> <p>この機能によって、次のコマンドが変更されました。</p> <ul style="list-style-type: none"> • redistribute (IP) <p>この機能によって、次のコマンドが追加されました。</p> <ul style="list-style-type: none"> • route-replicate (VRF アドレス ファミリ)

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at www.cisco.com/go/trademarks. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1005R)

このマニュアルで使用している IP アドレスおよび電話番号は、実際のアドレスおよび電話番号を示すものではありません。マニュアル内の例、コマンド出力、ネットワーク トポロジ図、およびその他の図は、説明のみを目的として使用されています。説明の中に実際のアドレスおよび電話番号が使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

© 2010 Cisco Systems, Inc.
All rights reserved.

Copyright © 2010–2011, シスコシステムズ合同会社.
All rights reserved.

