



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切なコマンドは、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.