



BGP ルータ ID の VRF 単位での割り当て

BGP ルータ ID の VRF 単位の割り当て機能により、同じルータ上のボーダー ゲートウェイ プロトコル (BGP) 内に VRF-to-VRF ピアリングを持つ機能が追加されます。BGP は、ルータ ID チェックのため、BGP 自身でセッションを拒否するように設計されています。VRF 単位の割り当て機能を使用すると、既存の **bgp router-id** コマンドの新しいキーワードを使用して、VRF 単位で異なるルータ ID を使用できます。ルータ ID は、VRF 単位での手動設定、または、アドレスファミリ コンフィギュレーション モードでのグローバルな自動割り当てや VRF 単位の自動割り当てが可能です。

- [機能情報の確認 \(1 ページ\)](#)
- [BGP ルータ ID の VRF 単位での割り当ての前提条件 \(2 ページ\)](#)
- [BGP ルータ ID の VRF 単位での割り当てに関する情報 \(2 ページ\)](#)
- [BGP ルータ ID の VRF 単位での割り当ての設定方法 \(3 ページ\)](#)
- [BGP ルータ ID の VRF 単位での割り当ての設定例 \(20 ページ\)](#)
- [その他の参考資料 \(27 ページ\)](#)
- [BGP ルータ ID の VRF 単位での割り当てに関する機能情報 \(28 ページ\)](#)

機能情報の確認

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

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

BGP ルータ ID の VRF 単位での割り当ての前提条件

この機能を設定する前に、ネットワーク内で Cisco Express Forwarding または distributed Cisco Express Forwarding (dCEF; 分散型シスコ エクスプレス フォワーディング) が有効になっている必要があります。また、BGP ピアリングがネットワーク内で実行されていることが前提になっています。

BGP ルータ ID の VRF 単位での割り当てに関する情報

BGP ルータ ID

BGP ルータ ID は、ルータの最大 IP アドレスに設定される 4 バイト フィールドです。ループバック インターフェイス アドレスは物理インターフェイスよりも安定しているため、ループバック インターフェイスのアドレスが物理インターフェイスよりも前に考慮されます。BGP ルータ ID は、最小ルータ ID を持つ BGP ルータにプリファレンスが設定されている宛先へのベストパスを決定するために、BGP アルゴリズムで使用されます。`bgp router-id` コマンドで BGP ルータ ID を手動で設定して、ベストパスのアルゴリズムに影響を与えることが可能です。

VRF 単位でのルータ ID の割り当て

Cisco IOS XE Release 2.1 以降のリリースでは、各バーチャルプライベートネットワーク (VPN) ルーティング/転送 (VRF) インスタンスに対する個別のルータ ID の設定に対するサポートが追加されました。BGP ルータ ID の VRF 単位での割り当て機能により、同じルータ上のボーダゲートウェイ プロトコル (BGP) 内に VRF-to-VRF ピアリングを持つ機能が追加されます。BGP は、ルータ ID チェックのため、BGP 自身でセッションを拒否するように設計されています。VRF 単位での割り当て機能を使用すると、既存の `bgp router-id` コマンドの新しいキーワードを使用して、VRF 単位で異なるルータ ID を使用できます。ルータ ID は、VRF 単位での手動設定、または、アドレス ファミリ コンフィギュレーション モードでのグローバルな自動割り当てや VRF 単位の自動割り当てが可能です。

ルート識別子

ルート識別子 (RD) はルーティングテーブルとフォワーディングテーブルを作成し、VPN のデフォルトのルート識別子を指定します。IPv4 プレフィックスをグローバルに固有の VPN-IPv4 プレフィックスに変更するために、RD が IPv4 プレフィックスの先頭に追加されます。RD は、自律システム番号と任意番号、または IP アドレスと任意番号のいずれかで構成できます。

RD は、次のいずれかの形式で入力できます。

- 16 ビット自律システム番号、コロン、32 ビット番号を入力します。次に例を示します。

45000:3

- 32 ビット IP アドレス、コロン、16 ビット番号を入力します。次に例を示します。

192.168.10.15:1

BGP ルータ ID の VRF 単位での割り当ての設定方法

VRF インスタンスの設定

VRF インスタンスを VRF 割り当て作業で使用されるように設定するには、この作業を実行します。この作業では、`vrf_trans` という名前の VRF インスタンスが作成されます。VRF を機能させるために、ルート識別子 (RD) が作成されます。ルート識別子が作成されると、`vrf_trans` という名前の VRF インスタンスにルーティングテーブルとフォワーディングテーブルが作成されます。

始める前に

この作業では、Cisco Express Forwarding または分散型 Cisco Express Forwarding が有効になっていることを前提としています。

手順の概要

1. `enable`
2. `configure terminal`
3. `ip vrf vrf-name`
4. `rd route-distinguisher`
5. `route-target {import | both} route-target-ext-community`
6. `route-target {export | both} route-target-ext-community`
7. `exit`
8. 定義する VRF 単位で、ステップ 3 ～ステップ 7 を繰り返します。

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： <pre>Router> enable</pre>	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例： <pre>Router# configure terminal</pre>	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	ip vrf <i>vrf-name</i> 例 : <pre>Router(config)# ip vrf vrf_trans</pre>	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	rd <i>route-distinguisher</i> 例 : <pre>Router(config-vrf)# rd 45000:2</pre>	<p>VRF にルーティング テーブルとフォワーディング テーブルを作成し、VPN にデフォルト RD を指定します。</p> <ul style="list-style-type: none"> VPN にデフォルト RD を指定するには、<i>route-distinguisher</i> 引数を使用します。RD の指定に使用できる形式は 2 つあります。詳細については、「ルート識別子」の項を参照してください。 この例では、RD は、コロンの後に番号 2 を持つ自律システム番号を使用します。
ステップ 5	route-target {import both} route-target-ext-community 例 : <pre>Router(config-vrf)# route-target import 55000:5</pre>	<p>VRF 用にルートターゲット拡張コミュニティを作成します。</p> <ul style="list-style-type: none"> ターゲット VPN 拡張コミュニティからルーティング情報をインポートするには、import キーワードを使用します。 ターゲット VPN 拡張コミュニティからルーティング情報をインポートするとともに、ルーティング情報を拡張コミュニティにエクスポートするには、both キーワードを使用します。 VPN 拡張コミュニティを指定するには、<i>route-target-ext-community</i> 引数を使用します。
ステップ 6	route-target {export both} route-target-ext-community 例 : <pre>Router(config-vrf)# route-target export 55000:1</pre>	<p>VRF 用にルートターゲット拡張コミュニティを作成します。</p> <ul style="list-style-type: none"> ターゲット VPN 拡張コミュニティにルーティング情報をエクスポートするには、export キーワードを使用します。 ターゲット VPN 拡張コミュニティからルーティング情報をインポートするとともに、ルーティング情報を拡張コミュニティにエクスポートするには、both キーワードを使用します。 VPN 拡張コミュニティを指定するには、<i>route-target-ext-community</i> 引数を使用します。

	コマンドまたはアクション	目的
ステップ 7	exit 例： Router(config-vrf)# exit	VRF コンフィギュレーションモードを終了し、グローバル コンフィギュレーションモードに戻ります。
ステップ 8	定義する VRF 単位で、ステップ 3 ～ステップ 7 を繰り返します。	--

VRF インスタンスとインターフェイスの関連付け

VRF 単位での割り当て作業で使用されるインターフェイスに VRF インスタンスを関連付けるには、この作業を実行します。この作業では、`vrf_trans` という名前の VRF インスタンスがシリアルインターフェイスに関連付けられます。

ip vrf forwarding コマンドにより IP アドレスが削除されるため、VRF インスタンスを関連付けるインターフェイスの IP アドレスをメモしておいてください。ステップ 8 で IP アドレスを再設定できます。

始める前に

- この作業では、Cisco Express Forwarding または分散型 Cisco Express Forwarding が有効になっていることを前提としています。
- この作業は、VRF インスタンスが [VRF インスタンスの設定 \(3 ページ\)](#) で設定されていることを前提としています。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask* [**secondary**]
5. **exit**
6. **interface** *type number*
7. **ip vrf forwarding** *vrf-name* [**downstream** *vrf-name2*]
8. **ip address** *ip-address mask* [**secondary**]
9. インターフェイスに関連付ける VRF 単位で、ステップ 5 ～ 8 を繰り返します。
10. **end**
11. **show ip vrf** [**brief** | **detail** | **interfaces** | **id**] [*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 モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： 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	interface type number 例： Router(config)# interface loopback0	インターフェイス タイプを設定し、インターフェイス コンフィギュレーションモードを開始します。 • この例では、ループバック インターフェイス 0 が設定されます。
ステップ 4	ip address ip-address mask [secondary] 例： Router(config-if)# ip address 172.16.1.1 255.255.255.255	IP アドレスを設定します。 • この例では、ループバック インターフェイス が 172.16.1.1 の IP アドレスで設定されます。
ステップ 5	exit 例： Router(config-if)# exit	インターフェイス コンフィギュレーション モードを終了し、グローバルコンフィギュレーションモードに戻ります。
ステップ 6	interface type number 例： Router(config)# interface serial2/0/0	インターフェイス タイプを設定し、インターフェイス コンフィギュレーションモードを開始します。 • この例では、シリアルインターフェイス 2/0/0 が設定されています。
ステップ 7	ip vrf forwarding vrf-name [downstream vrf-name2] 例： Router(config-if)# ip vrf forwarding vrf_trans	VRF をインターフェイスまたはサブインターフェイスと関連付けます。 • この例では、vrf_trans という名前の VRF がシリアルインターフェイス 2/0/0 に関連付けられます。 (注) インターフェイスにこのコマンドを実行すると、IP アドレスが削除されます。IP アドレスを再設定する必要があります。

	コマンドまたはアクション	目的
ステップ 8	ip address <i>ip-address mask [secondary]</i> 例 : <pre>Router(config-if)# ip address 192.168.4.1 255.255.255.0</pre>	IP アドレスを設定します。 <ul style="list-style-type: none"> この例では、シリアルインターフェイス 2/0/0 が 192.168.4.1 の IP アドレスで設定されます。
ステップ 9	インターフェイスに関連付ける VRF 単位で、ステップ 5～8 を繰り返します。	--
ステップ 10	end 例 : <pre>Router(config-if)# end</pre>	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 11	show ip vrf [brief detail interfaces id] [vrf-name] 例 : <pre>Router# show ip vrf interfaces</pre>	(任意) 定義された VRF および関連付けられたインターフェイスのセットを表示します。 <ul style="list-style-type: none"> この例では、このコマンド出力に、作成された VRF および関連付けられたインターフェイスが表示されます。

例

次の出力は、vrf_trans と vrf_users という名前の 2 つの VRF インスタンスが 2 つのシリアルインターフェイスに設定されたことを示しています。

```
Router# show ip vrf interfaces
Interface      IP-Address      VRF              Protocol
Serial2        192.168.4.1     vrf_trans        up
Serial3        192.168.5.1     vrf_user         up
```

VRF 単位での BGP ルータ ID の手動設定

VRF 単位で BGP ルータ ID を手動で設定するには、この作業を実行します。この作業では、複数のアドレスファミリー コンフィギュレーションが示され、1 つの VRF インスタンスに対して、IPv4 アドレス ファミリ モードでルータ ID が設定されます。ステップ 22 は、特定のステップを繰り返して、同じルータ上で複数の VRF の設定を許可する方法を示します。

始める前に

この作業は、事前に VRF インスタンスを作成し、そのインスタンスをインターフェイスに関連付けていることを前提とします。詳細については、[VRF インスタンスの設定 \(3 ページ\)](#) および [VRF インスタンスとインターフェイスの関連付け \(5 ページ\)](#) を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **router bgp** *autonomous-system-number*
4. **no bgp default ipv4-unicast**
5. **bgp log-neighbor-changes**
6. **neighbor** {*ip-address*| *peer-group-name*} **remote-as** *autonomous-system-number*
7. **neighbor** {*ip-address*| *peer-group-name*} **update-source** *interface-type interface-number*
8. **address-family** {**ipv4** [**mdt** | **multicast** | **unicast** [*vrf vrf-name*] | **vrf** *vrf-name*] | **vpn4** [**unicast**]}
9. **neighbor** {*ip-address*| *peer-group-name*} **activate**
10. **neighbor** {*ip-address*| *peer-group-name*} **send-community** {**both** | **standard** | **extended**}
11. **exit-address-family**
12. **address-family** {**ipv4** [**mdt** | **multicast** | **unicast** [*vrf vrf-name*] | **vrf** *vrf-name*] | **vpn4** [**unicast**]}
13. **redistribute connected**
14. **neighbor** {*ip-address*| *peer-group-name*} **remote-as** *autonomous-system-number*
15. **neighbor** *ip-address* **local-as** *autonomous-system-number* [**no-prepend** [**replace-as** [**dual-as**]]]
16. **neighbor** {*ip-address*| *peer-group-name*} **ebgp-multihop**[*tll*]
17. **neighbor** {*ip-address*| *peer-group-name*} **activate**
18. **neighbor** *ip-address* **allowas-in** [*number*]
19. **no auto-summary**
20. **no synchronization**
21. **bgp router-id** {*ip-address*| **auto-assign**}
22. 別の VRF インスタンスを設定するには、ステップ 11 ~ 21 を繰り返します。
23. **end**
24. **show ip bgp vpn4** {**all** | **rd** *route-distinguisher* | **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 モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： 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 bgp <i>autonomous-system-number</i> 例： Router(config)# router bgp 45000	指定したルーティング プロセスのルータ コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 4	no bgp default ipv4-unicast 例 : <pre>Router(config-router)# no bgp default ipv4-unicast</pre>	BGP ルーティングプロセスで使用される IPv4 ユニキャストアドレスファミリを無効にします。 (注) IPv4 ユニキャストアドレスファミリのルーティング情報は、 neighbor remote-as ルータコンフィギュレーションコマンドで設定された各 BGP ルーティングセッションに対して、デフォルトでアドバタイズされます。ただし、 neighbor remote-as コマンドを設定する前に、 no bgp default ipv4-unicast ルータコンフィギュレーションコマンドを設定した場合は例外です。既存のネイバーコンフィギュレーションは影響されません。
ステップ 5	bgp log-neighbor-changes 例 : <pre>Router(config-router)# bgp log-neighbor-changes</pre>	BGP ネイバーリセットのロギングを有効にします。
ステップ 6	neighbor {ip-address peer-group-name} remote-as autonomous-system-number 例 : <pre>Router(config-router)# neighbor 192.168.1.1 remote-as 45000</pre>	指定された自律システム内のネイバーの IP アドレスまたはピアグループ名を、ローカルルータの IPv4 マルチプロトコル BGP ネイバーテーブルに追加します。 <ul style="list-style-type: none"> • <i>autonomous-system-number</i> 引数が、router bgp コマンドで指定された自律システム番号と一致する場合、ネイバーは内部ネイバーになります。 • <i>autonomous-system-number</i> 引数が、router bgp コマンドで指定された自律システム番号と一致しない場合、ネイバーは外部ネイバーになります。 • この例では、ネイバーは内部ネイバーになります。
ステップ 7	neighbor {ip-address peer-group-name} update-source interface-type interface-number 例 : <pre>Router(config-router)# neighbor 192.168.1.1 update-source loopback0</pre>	BGP セッションが、TCP 接続の動作インターフェイスを使用できるようにします。 <ul style="list-style-type: none"> • この例では、指定されたネイバーの BGP TCP 接続が、最良のローカルアドレスではなく、ループバックインターフェイスの IP アドレスで発信されます。

	コマンドまたはアクション	目的
ステップ 8	address-family {ipv4 [mdt multicast unicast [vrf vrf-name] vrf vrf-name] vpnv4 [unicast]} 例 : <pre>Router(config-router)# address-family vpnv4</pre>	アドレスファミリー コンフィギュレーション モードを開始して、アドレスファミリー固有の設定を受け入れるよう BGP ピアを設定します。 <ul style="list-style-type: none"> この例では、VPNv4 アドレスファミリー セッションを作成します。
ステップ 9	neighbor {ip-address peer-group-name} activate 例 : <pre>Router(config-router-af)# neighbor 172.16.1.1 activate</pre>	VPNv4 アドレスファミリーの下のネイバーをアクティブにします。 <ul style="list-style-type: none"> この例では、ネイバー 172.16.1.1 がアクティブ化されます。
ステップ 10	neighbor {ip-address peer-group-name} send-community {both standard extended} 例 : <pre>Router(config-router-af)# neighbor 172.16.1.1 send-community extended</pre>	コミュニティ属性が BGP ネイバーに送信されるように指定します。 <ul style="list-style-type: none"> この例では、拡張コミュニティ属性が 172.16.1.1 のネイバーに送信されます。
ステップ 11	exit-address-family 例 : <pre>Router(config-router-af)# exit-address-family</pre>	アドレスファミリー コンフィギュレーション モードを終了し、ルータ コンフィギュレーション モードに戻ります。
ステップ 12	address-family {ipv4 [mdt multicast unicast [vrf vrf-name] vrf vrf-name] vpnv4 [unicast]} 例 : <pre>Router(config-router)# address-family ipv4 vrf vrf_trans</pre>	アドレスファミリー コンフィギュレーション モードを開始して、アドレスファミリー固有の設定を受け入れるよう BGP ピアを設定します。 <ul style="list-style-type: none"> この例では、vrf_trans という名前の VRF インスタンスが後続の IPv4 アドレスファミリー コンフィギュレーション コマンドに関連付けられるように指定します。
ステップ 13	redistribute connected 例 : <pre>Router(config-router-af)# redistribute connected</pre>	あるルーティング ドメインから別のルーティング ドメインに再配布します。 <ul style="list-style-type: none"> この例では、インターフェイスで IP が有効化されると自動的に確立されるルートを表すために、connected キーワードが使用されます。 この手順に適用される構文だけが表示されます。詳細については、『Cisco IOS IP Routing: BGP Command Reference』を参照してください。

	コマンドまたはアクション	目的
ステップ 14	<p>neighbor {<i>ip-address</i> <i>peer-group-name</i>} remote-as <i>autonomous-system-number</i></p> <p>例 :</p> <pre>Router(config-router-af)# neighbor 192.168.1.1 remote-as 40000</pre>	<p>指定された自律システム内のネイバーの IP アドレスまたはピア グループ名を、ローカルルータの IPv4 マルチプロトコル BGP ネイバーテーブルに追加します。</p> <ul style="list-style-type: none"> • <i>autonomous-system-number</i> 引数が、router bgp コマンドで指定された自律システム番号と一致する場合、ネイバーは内部ネイバーになります。 • <i>autonomous-system-number</i> 引数が、router bgp コマンドで指定された自律システム番号と一致しない場合、ネイバーは外部ネイバーになります。 • この例では、192.168.1.1 のネイバーは外部ネイバーです。
ステップ 15	<p>neighbor <i>ip-address</i> local-as <i>autonomous-system-number</i> [no-prepend [replace-as [dual-as]]]</p> <p>例 :</p> <pre>Router(config-router-af)# neighbor 192.168.1.1 local-as 50000 no-prepend</pre>	<p>eBGP ネイバーから受信したルートの AS_PATH 属性をカスタマイズします。</p> <ul style="list-style-type: none"> • ローカル BGP ルーティングプロセスからの自律システム番号は、デフォルトで、すべての外部ルートに追加されます。 • eBGP ネイバーから受信されたルートにローカル自律システム番号を付加しない場合は、no-prepend キーワードを使用します。 • この例では、192.168.1.1 のネイバーからのルートにローカル自律システム番号が含まれていません。
ステップ 16	<p>neighbor {<i>ip-address</i> <i>peer-group-name</i>} ebgp-multihop[<i>ttl</i>]</p> <p>例 :</p> <pre>Router(config-router-af)# neighbor 192.168.1.1 ebgp-multihop 2</pre>	<p>直接接続されていないネットワーク上の外部ピアからの BGP 接続を受け入れ、またそのピアへの BGP 接続を試みます。</p> <ul style="list-style-type: none"> • この例では、直接接続されていないネットワーク上に存在するネイバー 192.168.1.1 との接続ができるように BGP を設定します。
ステップ 17	<p>neighbor {<i>ip-address</i> <i>peer-group-name</i>} activate</p> <p>例 :</p> <pre>Router(config-router-af)# neighbor 192.168.1.1 activate</pre>	<p>このネイバーを IPv4 アドレスファミリの下でアクティブ化します。</p> <ul style="list-style-type: none"> • この例では、ネイバー 192.168.1.1 がアクティブにされます。

	コマンドまたはアクション	目的
ステップ 18	neighbor ip-address allowas-in [number] 例 : <pre>Router(config-router-af)# neighbor 192.168.1.1 allowas-in 1</pre>	複製の自律システム番号が含まれるプレフィックスをすべて再アドバタイズできるように、プロバイダー エッジ (PE) ルータを設定します。 <ul style="list-style-type: none"> この例では、自律システム番号が 45000 の PE ルータが VRF vrf-trans からのプレフィックスを許可するように設定されます。IP アドレスが 192.168.1.1 のネイバー PE ルータが、同じ自律システム番号の別の PE ルータに 1 回再アドバタイズされるように設定されます。
ステップ 19	no auto-summary 例 : <pre>Router(config-router-af)# no auto-summary</pre>	自動サマライズを無効にし、サブプレフィックスルーティング情報をクラスフル ネットワーク境界間で送信します。
ステップ 20	no synchronization 例 : <pre>Router(config-router-af)# no synchronization</pre>	Cisco IOS XE ソフトウェアが内部ゲートウェイ プロトコル (IGP) との同期を待たずにネットワーク ルートをアドバタイズできるようにします。
ステップ 21	bgp router-id {ip-address auto-assign} 例 : <pre>Router(config-router-af)# bgp router-id 10.99.1.1</pre>	ローカル BGP ルーティングプロセスの固定ルータ ID を設定します。 <ul style="list-style-type: none"> この例では、指定された BGP ルータ ID が、IPv4 アドレス ファミリ コンフィギュレーションに関連付けられた VRF インスタンスに割り当てられます。
ステップ 22	別の VRF インスタンスを設定するには、ステップ 11 ~ 21 を繰り返します。	--
ステップ 23	end 例 : <pre>Router(config-router-af)# end</pre>	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て、特権 EXEC モードに戻ります。
ステップ 24	show ip bgp vpnv4 {all rd route-distinguisher vrf vrf-name} 例 : <pre>Router# show ip bgp vpnv4 all</pre>	(任意) BGP テーブルからの VPN アドレス情報を表示します。 <ul style="list-style-type: none"> この例では、すべての VPNv4 データベースが表示されます。

	コマンドまたはアクション	目的
		(注) この例では、このタスクに適用可能な構文だけが使用されています。詳細については、『Cisco IOS Multiprotocol Label Switching Command Reference』を参照してください。

例

次のサンプル出力は、vrf_trans と vrf_user という名前の 2 つの VRF インスタンスが個別のルータ ID で設定されていることを前提としています。ルータ ID が VRF 名の次に表示されます。

```
Router# show ip bgp vpnv4 all
BGP table version is 5, local router ID is 172.17.1.99
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop          Metric LocPrf Weight Path
Route Distinguisher: 1:1 (default for vrf vrf_trans) VRF Router ID 10.99.1.2
*> 192.168.4.0      0.0.0.0            0          32768 ?
Route Distinguisher: 42:1 (default for vrf vrf_user) VRF Router ID 10.99.1.1
*> 192.168.5.0      0.0.0.0            0          32768 ?
```

VRF 単位での BGP ルータ ID の自動割り当て

VRF 単位で BGP ルータ ID を自動で設定するには、この作業を実行します。この作業では、ループバック インターフェイスが VRF に関連付けられ、**bgp router-id** コマンドがルータ コンフィギュレーション レベルで設定されて、BGP ルータ ID がすべての VRF インスタンスに自動的に割り当てられます。ステップ 9 は、特定のステップを繰り返して、インターフェイスに関連付けられる各 VRF を設定する方法を示します。ステップ 30 は、同じルータ上で複数の VRF を設定する方法を示します。

始める前に

この作業は、事前に VRF インスタンスを作成していることを前提とします。詳細については、[VRF インスタンスの設定 \(3 ページ\)](#) を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask* [**secondary**]
5. **exit**
6. **interface** *type number*

7. **ip vrf forwarding** *vrf-name* [**downstream** *vrf-name2*]
8. **ip address** *ip-address mask* [**secondary**]
9. インターフェイスに関連付ける VRF 単位で、ステップ 5 ~ 8 を繰り返します。
10. **exit**
11. **router bgp** *autonomous-system-number*
12. **bgp router-id** {*ip-address*| **vrf auto-assign**}
13. **no bgp default ipv4-unicast**
14. **bgp log-neighbor-changes**
15. **neighbor** {*ip-address*| *peer-group-name*} **remote-as** *autonomous-system-number*
16. **neighbor** {*ip-address*| *peer-group-name*} **update-source** *interface-type interface-number*
17. **address-family** {**ipv4** [**mdt** | **multicast** | **unicast** [*vrf vrf-name*] | **vrf** *vrf-name*] | **vpn4** [**unicast**]}
18. **neighbor** {*ip-address*| *peer-group-name*} **activate**
19. **neighbor** {*ip-address*| *peer-group-name*} **send-community** {**both** | **standard** | **extended**}
20. **exit-address-family**
21. **address-family** {**ipv4** [**mdt** | **multicast** | **unicast** [*vrf vrf-name*] | **vrf** *vrf-name*] | **vpn4** [**unicast**]}
22. **redistribute** **connected**
23. **neighbor** {*ip-address*| *peer-group-name*} **remote-as** *autonomous-system-number*
24. **neighbor** *ip-address* **local-as** *autonomous-system-number* [**no-prepend** [**replace-as** [*dual-as*]]]
25. **neighbor** {*ip-address*| *peer-group-name*} **ebgp-multihop**[*ttl*]
26. **neighbor** {*ip-address*| *peer-group-name*} **activate**
27. **neighbor** *ip-address* **allowas-in** [*number*]
28. **no auto-summary**
29. **no synchronization**
30. 別の VRF インスタンスを設定するには、ステップ 20 ~ 29 を繰り返します。
31. **end**
32. **show ip bgp vpn4** {**all** | **rd** *route-distinguisher* | **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 モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： 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	interface <i>type number</i> 例： Router(config)# interface loopback0	インターフェイス タイプを設定し、インターフェイス コンフィギュレーションモードを開始します。 • この例では、ループバック インターフェイス 0 が設定されます。

	コマンドまたはアクション	目的
ステップ 4	ip address <i>ip-address mask</i> [secondary] 例 : <pre>Router(config-if)# ip address 172.16.1.1 255.255.255.255</pre>	IP アドレスを設定します。 <ul style="list-style-type: none"> この例では、ループバック インターフェイスが 172.16.1.1 の IP アドレスで設定されます。
ステップ 5	exit 例 : <pre>Router(config-if)# exit</pre>	インターフェイス コンフィギュレーション モードを終了し、グローバルコンフィギュレーションモードに戻ります。
ステップ 6	interface <i>type number</i> 例 : <pre>Router(config)# interface loopback1</pre>	インターフェイス タイプを設定し、インターフェイス コンフィギュレーションモードを開始します。 <ul style="list-style-type: none"> この例では、ループバック インターフェイス 1 が設定されます。
ステップ 7	ip vrf forwarding <i>vrf-name</i> [downstream vrf-name2] 例 : <pre>Router(config-if)# ip vrf forwarding vrf_trans</pre>	VRF をインターフェイスまたはサブインターフェイスと関連付けます。 <ul style="list-style-type: none"> この例では、vrf_trans という名前の VRF がループバック インターフェイス 1 に関連付けられます。 (注) インターフェイスにこのコマンドを実行すると、IP アドレスが削除されます。IP アドレスを再設定する必要があります。
ステップ 8	ip address <i>ip-address mask</i> [secondary] 例 : <pre>Router(config-if)# ip address 10.99.1.1 255.255.255.255</pre>	IP アドレスを設定します。 <ul style="list-style-type: none"> この例では、ループバック インターフェイス 1 が 10.99.1.1 の IP アドレスで設定されます。
ステップ 9	インターフェイスに関連付ける VRF 単位で、ステップ 5 ~ 8 を繰り返します。	--
ステップ 10	exit 例 : <pre>Router(config-if)# exit</pre>	インターフェイス コンフィギュレーション モードを終了し、グローバルコンフィギュレーションモードに戻ります。
ステップ 11	router bgp <i>autonomous-system-number</i> 例 : <pre>Router(config)# router bgp 45000</pre>	指定したルーティング プロセスのルータ コンフィギュレーションモードを開始します。

	コマンドまたはアクション	目的
ステップ 12	bgp router-id {ip-address vrf auto-assign} 例 : <pre>Router(config-router)# bgp router-id vrf auto-assign</pre>	ローカル BGP ルーティングプロセスの固定ルータ ID を設定します。 <ul style="list-style-type: none"> この例では、BGP ルータ ID が VRF インスタンス単位で自動的に割り当てられます。
ステップ 13	no bgp default ipv4-unicast 例 : <pre>Router(config-router)# no bgp default ipv4-unicast</pre>	BGP ルーティングプロセスで使用される IPv4 ユニキャストアドレス ファミリを無効にします。 (注) IPv4 ユニキャストアドレス ファミリのルーティング情報は、 neighbor remote-as ルータ コンフィギュレーション コマンドで設定された各 BGP ルーティングセッションに対して、デフォルトでアドバタイズされます。ただし、 neighbor remote-as コマンドを設定する前に、 no bgp default ipv4-unicast ルータ コンフィギュレーション コマンドを設定した場合は例外です。既存のネイバー コンフィギュレーションは影響されません。
ステップ 14	bgp log-neighbor-changes 例 : <pre>Router(config-router)# bgp log-neighbor-changes</pre>	BGP ネイバーリセットのロギングを有効にします。
ステップ 15	neighbor {ip-address peer-group-name} remote-as autonomous-system-number 例 : <pre>Router(config-router)# neighbor 192.168.1.1 remote-as 45000</pre>	指定された自律システム内のネイバーの IP アドレスまたはピア グループ名を、ローカルルータの IPv4 マルチプロトコル BGP ネイバーテーブルに追加します。 <ul style="list-style-type: none"> autonomous-system-number 引数が、router bgp コマンドで指定された自律システム番号と一致する場合、ネイバーは内部ネイバーになります。 autonomous-system-number 引数が、router bgp コマンドで指定された自律システム番号と一致しない場合、ネイバーは外部ネイバーになります。 この例では、ネイバーは内部ネイバーになります。

	コマンドまたはアクション	目的
ステップ 16	<p>neighbor {<i>ip-address</i> <i>peer-group-name</i>} update-source <i>interface-type interface-number</i></p> <p>例 :</p> <pre>Router(config-router)# neighbor 192.168.1.1 update-source loopback0</pre>	<p>BGP セッションが、TCP 接続の動作インターフェイスを使用できるようにします。</p> <ul style="list-style-type: none"> この例では、指定されたネイバーの BGP TCP 接続が、最良のローカルアドレスではなく、ループバック インターフェイスの IP アドレスで発信されます。
ステップ 17	<p>address-family {<i>ipv4</i> [<i>mdt</i> <i>multicast</i> <i>unicast</i> [<i>vrf vrf-name</i>] <i>vrf vrf-name</i>] <i>vpn4</i> [<i>unicast</i>]}]</p> <p>例 :</p> <pre>Router(config-router)# address-family vpn4</pre>	<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て、アドレスファミリ固有の設定を受け入れるよう BGP ピアを設定します。</p> <ul style="list-style-type: none"> この例では、VPNv4 アドレスファミリ セッションを作成します。
ステップ 18	<p>neighbor {<i>ip-address</i> <i>peer-group-name</i>} activate</p> <p>例 :</p> <pre>Router(config-router-af)# neighbor 172.16.1.1 activate</pre>	<p>VPNv4 アドレスファミリ の下のネイバーをアクティブにします。</p> <ul style="list-style-type: none"> この例では、ネイバー 172.16.1.1 がアクティブ化されます。
ステップ 19	<p>neighbor {<i>ip-address</i> <i>peer-group-name</i>} send-community {<i>both</i> <i>standard</i> <i>extended</i>}</p> <p>例 :</p> <pre>Router(config-router-af)# neighbor 172.16.1.1 send-community extended</pre>	<p>コミュニティ属性が BGP ネイバーに送信されるように指定します。</p> <ul style="list-style-type: none"> この例では、拡張コミュニティ属性が 172.16.1.1 のネイバーに送信されます。
ステップ 20	<p>exit-address-family</p> <p>例 :</p> <pre>Router(config-router-af)# exit-address-family</pre>	<p>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、ルータ コンフィギュレーションモードに戻ります。</p>
ステップ 21	<p>address-family {<i>ipv4</i> [<i>mdt</i> <i>multicast</i> <i>unicast</i> [<i>vrf vrf-name</i>] <i>vrf vrf-name</i>] <i>vpn4</i> [<i>unicast</i>]}]</p> <p>例 :</p> <pre>Router(config-router)# address-family ipv4 vrf vrf_trans</pre>	<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て、アドレスファミリ固有の設定を受け入れるよう BGP ピアを設定します。</p> <ul style="list-style-type: none"> この例では、<i>vrf_trans</i> という名前の VRF インスタンスが後続の IPv4 アドレスファミリ コンフィギュレーションモードのコマンドに関連付けられるように指定します。
ステップ 22	<p>redistribute connected</p> <p>例 :</p>	<p>あるルーティング ドメインから別のルーティング ドメインに再配布します。</p>

	コマンドまたはアクション	目的
	<pre>Router(config-router-af)# redistribute connected</pre>	<ul style="list-style-type: none"> この例では、インターフェイスで IP が有効化されると自動的に確立されるルートを表すために、connected キーワードが使用されます。 この手順に適用される構文だけが表示されます。詳細については、『<i>Cisco IOS IP Routing: BGP Command Reference</i>』を参照してください。
ステップ 23	<p>neighbor {<i>ip-address</i> <i>peer-group-name</i>} remote-as <i>autonomous-system-number</i></p> <p>例 :</p> <pre>Router(config-router-af)# neighbor 192.168.1.1 remote-as 40000</pre>	<p>指定された自律システム内のネイバーの IP アドレスまたはピア グループ名を、ローカルルータの IPv4 マルチプロトコル BGP ネイバーテーブルに追加します。</p> <ul style="list-style-type: none"> <i>autonomous-system-number</i> 引数が、router bgp コマンドで指定された自律システム番号と一致する場合、ネイバーは内部ネイバーになります。 <i>autonomous-system-number</i> 引数が、router bgp コマンドで指定された自律システム番号と一致しない場合、ネイバーは外部ネイバーになります。 この例では、192.168.1.1 のネイバーは外部ネイバーです。
ステップ 24	<p>neighbor <i>ip-address</i> local-as <i>autonomous-system-number</i> [no-prepend [replace-as [dual-as]]]</p> <p>例 :</p> <pre>Router(config-router-af)# neighbor 192.168.1.1 local-as 50000 no-prepend</pre>	<p>eBGP ネイバーから受信したルートの AS_PATH 属性をカスタマイズします。</p> <ul style="list-style-type: none"> ローカル BGP ルーティングプロセスからの自律システム番号は、デフォルトで、すべての外部ルートに追加されます。 eBGP ネイバーから受信されたルートにローカル自律システム番号を付加しない場合は、no-prepend キーワードを使用します。 この例では、192.168.1.1 のネイバーからのルートにローカル自律システム番号が含まれていません。
ステップ 25	<p>neighbor {<i>ip-address</i> <i>peer-group-name</i>} ebgp-multihop[<i>ttl</i>]</p> <p>例 :</p>	<p>直接接続されていないネットワーク上の外部ピアからの BGP 接続を受け入れ、またそのピアへの BGP 接続を試みます。</p>

	コマンドまたはアクション	目的
	Router(config-router-af)# neighbor 192.168.1.1 ebgp-multihop 2	<ul style="list-style-type: none"> この例では、直接接続されていないネットワーク上に存在するネイバー 192.168.1.1 との接続ができるように BGP を設定します。
ステップ 26	neighbor {ip-address peer-group-name} activate 例 : Router(config-router-af)# neighbor 192.168.1.1 activate	このネイバーを IPv4 アドレス ファミリの下でアクティブ化します。 <ul style="list-style-type: none"> この例では、ネイバー 192.168.1.1 がアクティブにされます。
ステップ 27	neighbor ip-address allowas-in [number] 例 : Router(config-router-af)# neighbor 192.168.1.1 allowas-in 1	複製の自律システム番号が含まれるプレフィックスをすべて再アドバタイズできるように、プロバイダー エッジ (PE) ルータを設定します。 <ul style="list-style-type: none"> この例では、自律システム番号が 45000 の PE ルータが VRF vrf-trans からのプレフィックスを許可するように設定されます。IP アドレスが 192.168.1.1 のネイバー PE ルータが、同じ自律システム番号の別の PE ルータに 1 回再アドバタイズされるように設定されます。
ステップ 28	no auto-summary 例 : Router(config-router-af)# no auto-summary	自動サマライズを無効にし、サブプレフィックスルーティング情報をクラスフル ネットワーク境界間で送信します。
ステップ 29	no synchronization 例 : Router(config-router-af)# no synchronization	Cisco IOS XE ソフトウェアが内部ゲートウェイ プロトコル (IGP) との同期を待たずにネットワーク ルートをアドバタイズできるようにします。
ステップ 30	別の VRF インスタンスを設定するには、ステップ 20 ~ 29 を繰り返します。	--
ステップ 31	end 例 : Router(config-router-af)# end	アドレス ファミリ コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。
ステップ 32	show ip bgp vpnv4 {all rd route-distinguisher vrf vrf-name} 例 : Router# show ip bgp vpnv4 all	(任意) BGP テーブルからの VPN アドレス情報を表示します。 <ul style="list-style-type: none"> この例では、すべての VPNv4 データベースが表示されます。

	コマンドまたはアクション	目的
		(注) この例では、このタスクに適用可能な構文だけが使用されています。詳細については、『Cisco IOS Multiprotocol Label Switching Command Reference』を参照してください。

例

次のサンプル出力は、vrf_trans と vrf_user という名前の 2 つの VRF インスタンスが個別のルータ ID で設定されていることを前提としています。ルータ ID が VRF 名の次に表示されます。

```
Router# show ip bgp vpnv4 all
BGP table version is 43, local router ID is 172.16.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network        Next Hop           Metric LocPrf Weight Path
Route Distinguisher: 1:1 (default for vrf vrf_trans) VRF Router ID 10.99.1.2
*> 172.22.0.0      0.0.0.0             0           32768 ?
r> 172.23.0.0      172.23.1.1          0           0 3 1 ?
*>i10.21.1.1/32    192.168.3.1         0    100      0 2 i
*> 10.52.1.0/24    172.23.1.1          0           0 3 1 ?
*> 10.52.2.1/32    172.23.1.1          0           0 3 1 3 i
*> 10.52.3.1/32    172.23.1.1          0           0 3 1 3 i
*> 10.99.1.1/32    172.23.1.1          0           0 3 1 ?
*> 10.99.1.2/32    0.0.0.0             0           32768 ?
Route Distinguisher: 10:1
*>i10.21.1.1/32    192.168.3.1         0    100      0 2 i
Route Distinguisher: 42:1 (default for vrf vrf_user) VRF Router ID 10.99.1.1
r> 172.22.0.0      172.22.1.1          0           0 2 1 ?
*> 172.23.0.0      0.0.0.0             0           32768 ?
*> 10.21.1.1/32    172.22.1.1          0           0 2 1 2 i
*>i10.52.1.0/24    192.168.3.1         0    100      0 ?
*>i10.52.2.1/32    192.168.3.1         0    100      0 3 i
*>i10.52.3.1/32    192.168.3.1         0    100      0 3 i
*> 10.99.1.1/32    0.0.0.0             0           32768 ?
*> 10.99.1.2/32    172.22.1.1          0           0 2 1 ?
```

BGP ルータ ID の VRF 単位での割り当ての設定例

VRF 単位での BGP ルータ ID の手動設定例

次の例は、vrf_trans と vrf_user の 2 つの VRF を、同じルータ上で相互間のセッションで設定する方法を示します。VRF 単位での BGP ルータ ID は、個別の IPv4 アドレス ファミリの下で手動で設定されます。show ip bgp vpnv4 コマンドを使用すると、ルータ ID が VRF 単位に設定さ

れていることを確認できます。このコンフィギュレーションは、グローバルコンフィギュレーションモードで開始されます。

```
ip vrf vrf_trans
 rd 45000:1
  route-target export 50000:50
  route-target import 40000:1
!
ip vrf vrf_user
 rd 65500:1
  route-target export 65500:1
  route-target import 65500:1
!
interface Loopback0
 ip address 10.1.1.1 255.255.255.255
!
router bgp 45000
 no bgp default ipv4-unicast
 bgp log-neighbor-changes
 neighbor 192.168.3.1 remote-as 45000
 neighbor 192.168.3.1 update-source Loopback0
!
 address-family vpnv4
  neighbor 192.168.3.1 activate
  neighbor 192.168.3.1 send-community extended
  exit-address-family
!
 address-family ipv4 vrf vrf_user
  redistribute connected
  neighbor 172.22.1.1 remote-as 40000
  neighbor 172.22.1.1 local-as 50000 no-prepend
  neighbor 172.22.1.1 ebgp-multihop 2
  neighbor 172.22.1.1 activate
  neighbor 172.22.1.1 allowas-in 1
  no auto-summary
  no synchronization
  bgp router-id 10.99.1.1
  exit-address-family
!
 address-family ipv4 vrf vrf_trans
  redistribute connected
  neighbor 172.23.1.1 remote-as 50000
  neighbor 172.23.1.1 local-as 40000 no-prepend
  neighbor 172.23.1.1 ebgp-multihop 2
  neighbor 172.23.1.1 activate
  neighbor 172.23.1.1 allowas-in 1
  no auto-summary
  no synchronization
  bgp router-id 10.99.1.2
  exit-address-family
```

コンフィギュレーションの後、**show ip bgp vpnv4 all** コマンドの出力には、VRF 名の次に表示されるルータ ID が表示されます。

```
Router# show ip bgp vpnv4 all
BGP table version is 43, local router ID is 10.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop          Metric LocPrf Weight Path
Route Distinguisher: 45000:1 (default for vrf vrf_trans) VRF Router ID 10.99.1.2
*> 172.22.0.0        0.0.0.0            0           32768 ?
```

VRF 単位での BGP ルータ ID の手動設定例

```

r> 172.23.0.0          172.23.1.1          0          0 3 1 ?
*>i10.21.1.1/32      192.168.3.1          0 100      0 2 i
*> 10.52.1.0/24      172.23.1.1          0 3 1 ?
*> 10.52.2.1/32      172.23.1.1          0 3 1 3 i
*> 10.52.3.1/32      172.23.1.1          0 3 1 3 i
*> 10.99.1.1/32      172.23.1.1          0          0 3 1 ?
*> 10.99.2.2/32      0.0.0.0             0          32768 ?
Route Distinguisher: 50000:1
*>i10.21.1.1/32      192.168.3.1          0 100      0 2 i
Route Distinguisher: 65500:1 (default for vrf vrf_user) VRF Router ID 10.99.1.1
r> 172.22.0.0          172.22.1.1          0          0 2 1 ?
*> 172.23.0.0          0.0.0.0             0          32768 ?
*> 10.21.1.1/32      172.22.1.1          0 2 1 2 i
*>i10.52.1.0/24      192.168.3.1          0 100      0 ?
*>i10.52.2.1/32      192.168.3.1          0 100      0 3 i
*>i10.52.3.1/32      192.168.3.1          0 100      0 3 i
*> 10.99.1.1/32      0.0.0.0             0          32768 ?
*> 10.99.2.2/32      172.22.1.1          0          0 2 1 ?

```

指定された VRF の **show ip bgp vpnv4 vrf** コマンドの出力には、出力ヘッダーにルータ ID が表示されます。

```

Router# show ip bgp vpnv4 vrf vrf_user
BGP table version is 43, local router ID is 10.99.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network        Next Hop        Metric LocPrf Weight Path
Route Distinguisher: 65500:1 (default for vrf vrf_user) VRF Router ID 10.99.1.1
r> 172.22.0.0      172.22.1.1      0          0 2 1 ?
*> 172.23.0.0      0.0.0.0         0          32768 ?
*> 10.21.1.1/32    172.22.1.1      0 2 1 2 i
*>i10.52.1.0/24    192.168.3.1     0 100      0 ?
*>i10.52.2.1/32    192.168.3.1     0 100      0 3 i
*>i10.52.3.1/32    192.168.3.1     0 100      0 3 i
*> 10.99.1.1/32    0.0.0.0         0          32768 ?
*> 10.99.2.2/32    172.22.1.1     0          0 2 1 ?

```

指定された VRF の **show ip bgp vpnv4 vrf summary** コマンドの出力には、出力の最初の行にルータ ID が表示されます。

```

Router# show ip bgp vpnv4 vrf vrf_user summary
BGP router identifier 10.99.1.1, local AS number 45000
BGP table version is 43, main routing table version 43
8 network entries using 1128 bytes of memory
8 path entries using 544 bytes of memory
16/10 BGP path/bestpath attribute entries using 1856 bytes of memory
6 BGP AS-PATH entries using 144 bytes of memory
3 BGP extended community entries using 72 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 3744 total bytes of memory
BGP activity 17/0 prefixes, 17/0 paths, scan interval 15 secs
Neighbor      V      AS MsgRcvd MsgSent  TblVer  InQ  OutQ  Up/Down  State/PfxRcd
172.22.1.1    4        2     20     21     43   0    0 00:12:33      3

```

パスが VRF で送信されると、指定された VRF とネットワーク アドレスの **show ip bgp vpnv4 vrf** コマンドの出力に、正しいルータ ID が表示されます。

```

Router# show ip bgp vpnv4 vrf vrf_user 172.23.0.0
BGP routing table entry for 65500:1:172.23.0.0/8, version 22

```

```

Paths: (1 available, best #1, table vrf_user)
  Advertised to update-groups:
    2          3
Local
  0.0.0.0 from 0.0.0.0 (10.99.1.1)
    Origin incomplete, metric 0, localpref 100, weight 32768, valid, sourced, best
    Extended Community: RT:65500:1

```

VRF 単位での BGP ルータ ID の自動割り当て例

次に、BGP が個別のルータ ID を各 VRF インスタンスに自動的に割り当てるように設定する 3 つの異なる設定例を示します。

ループバック インターフェイス IP アドレスを使用してグローバルに自動割り当てされるルータ ID の例

次の例は、vrf_trans と vrf_user の 2 つの VRF を、同じルータ上で相互間のセッションで設定する方法を示します。ルータ コンフィギュレーションモードでは、BGP が、各 VRF に BGP ルータ ID を自動的に割り当てるようにグローバルに設定されます。ループバック インターフェイスは、ルータ ID の IP アドレスを送信するために個別の VRF に関連付けられます。**show ip bgp vpnv4** コマンドを使用すると、ルータ ID が VRF 単位に設定されていることを確認できます。

```

ip vrf vrf_trans
  rd 45000:1
  route-target export 50000:50
  route-target import 40000:1
!
ip vrf vrf_user
  rd 65500:1
  route-target export 65500:1
  route-target import 65500:1
!
interface Loopback0
  ip address 10.1.1.1 255.255.255.255
!
interface Loopback1
  ip vrf forwarding vrf_user
  ip address 10.99.1.1 255.255.255.255
!
interface Loopback2
  ip vrf forwarding vrf_trans
  ip address 10.99.2.2 255.255.255.255
!
router bgp 45000
  bgp router-id vrf auto-assign
  no bgp default ipv4-unicast
  bgp log-neighbor-changes
  neighbor 192.168.3.1 remote-as 45000
  neighbor 192.168.3.1 update-source Loopback0
!
address-family vpnv4
  neighbor 192.168.3.1 activate
  neighbor 192.168.3.1 send-community extended
  exit-address-family
!
address-family ipv4 vrf vrf_user
  redistribute connected

```

デフォルト ルータ ID がない場合にグローバルに自動割り当てされるルータ ID の例

```

neighbor 172.22.1.1 remote-as 40000
neighbor 172.22.1.1 local-as 50000 no-prepend
neighbor 172.22.1.1 ebgp-multihop 2
neighbor 172.22.1.1 activate
neighbor 172.22.1.1 allowas-in 1
no auto-summary
no synchronization
exit-address-family
!
address-family ipv4 vrf vrf_trans
redistribute connected
neighbor 172.23.1.1 remote-as 50000
neighbor 172.23.1.1 local-as 2 no-prepend
neighbor 172.23.1.1 ebgp-multihop 2
neighbor 172.23.1.1 activate
neighbor 172.23.1.1 allowas-in 1
no auto-summary
no synchronization
exit-address-family

```

コンフィギュレーションの後、**show ip bgp vpnv4 all** コマンドの出力には、VRF 名の次に表示されるルータ ID が表示されます。この例で使用されているルータ ID が、ループバック インターフェイス 1 およびループバック インターフェイス 2 で設定された IP アドレスから送信されていることに注意してください。ルータ ID は、[VRF 単位での BGP ルータ ID の手動設定例 \(20 ページ\)](#) と同じです。

```

Router# show ip bgp vpnv4 all
BGP table version is 43, local router ID is 10.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop          Metric LocPrf Weight Path
Route Distinguisher: 45000:1 (default for vrf vrf_trans) VRF Router ID 10.99.2.2
*> 172.22.0.0        0.0.0.0            0          32768 ?
r> 172.23.0.0        172.23.1.1         0           0 3 1 ?
*>i10.21.1.1/32      192.168.3.1        0          100     0 2 i
*> 10.52.1.0/24      172.23.1.1         0           0 3 1 ?
*> 10.52.2.1/32      172.23.1.1         0           0 3 1 3 i
*> 10.52.3.1/32      172.23.1.1         0           0 3 1 3 i
*> 10.99.1.1/32      172.23.1.1         0           0 3 1 ?
*> 10.99.1.2/32      0.0.0.0            0          32768 ?
Route Distinguisher: 50000:1
*>i10.21.1.1/32      192.168.3.1        0          100     0 2 i
Route Distinguisher: 65500:1 (default for vrf vrf_user) VRF Router ID 10.99.1.1
r> 172.22.0.0        172.22.1.1         0           0 2 1 ?
*> 172.23.0.0        0.0.0.0            0          32768 ?
*> 10.21.1.1/32      172.22.1.1         0           0 2 1 2 i
*>i10.52.1.0/24      192.168.3.1        0          100     0 ?
*>i10.52.2.1/32      192.168.3.1        0          100     0 3 i
*>i10.52.3.1/32      192.168.3.1        0          100     0 3 i
*> 10.99.1.1/32      0.0.0.0            0          32768 ?
*> 10.99.1.2/32      172.22.1.1         0           0 2 1 ?

```

デフォルト ルータ ID がない場合にグローバルに自動割り当てされるルータ ID の例

次に、ルータを設定して、デフォルトのルータ ID が割り当てられない場合に自動的に BGP ルータ ID が割り当てられる VRF を関連付ける例を示します。

```

ip vrf vpn1
rd 45000:1

```

```

route-target export 45000:1
route-target import 45000:1
!
interface Loopback0
 ip vrf forwarding vpn1
 ip address 10.1.1.1 255.255.255.255
!
router bgp 45000
 bgp router-id vrf auto-assign
 no bgp default ipv4-unicast
 bgp log-neighbor-changes
!
 address-family ipv4 vrf vpn1
  neighbor 172.22.1.2 remote-as 40000
  neighbor 172.22.1.2 activate
 no auto-summary
 no synchronization
 exit-address-family

```

別のルータが 2 つのルータ間のセッションを確立するように設定されていることを前提として、**show ip interface brief** コマンドの出力には、設定済みの VRF インターフェイスだけが表示されます。

```

Router# show ip interface brief
Interface          IP-Address      OK? Method Status              Protocol
Serial2/0/0        unassigned      YES NVRAM   administratively down down
Serial3/0/0        unassigned      YES NVRAM   administratively down down
Loopback0          10.1.1.1        YES NVRAM   up                  up

```

show ip vrf コマンドを使用すると、ルータ ID が VRF に対して割り当てられていることを確認できます。

```

Router# show ip vrf
Name                Default RD      Interfaces
vpn1                 45000:1        Loopback0
VRF session is established:

```

VRF 単位で自動割り当てされるルータ ID の例

次の例は、`vrf_trans` と `vrf_user` の 2 つの VRF を、同じルータ上で相互間のセッションで設定する方法を示します。個別の VRF に関連付けられた IPv4 アドレス ファミリの下では、BGP が自動的に BGP ルータ ID を割り当てるように設定されます。ループバック インターフェイスは、ルータ ID の IP アドレスを送信するために個別の VRF に関連付けられます。**show ip bgp vpv4** コマンドを使用すると、ルータ ID が VRF 単位に設定されていることを確認できます。

```

ip vrf vrf_trans
 rd 45000:1
 route-target export 50000:50
 route-target import 40000:1
!
ip vrf vrf_user
 rd 65500:1
 route-target export 65500:1
 route-target import 65500:1
!
interface Loopback0
 ip address 10.1.1.1 255.255.255.255
!

```

VRF 単位で自動割り当てされるルータ ID の例

```

interface Loopback1
 ip vrf forwarding vrf_user
 ip address 10.99.1.1 255.255.255.255
!
interface Loopback2
 ip vrf forwarding vrf_trans
 ip address 10.99.2.2 255.255.255.255
!
router bgp 45000
 no bgp default ipv4-unicast
 bgp log-neighbor-changes
 neighbor 192.168.3.1 remote-as 45000
 neighbor 192.168.3.1 update-source Loopback0
!
address-family vpnv4
 neighbor 192.168.3.1 activate
 neighbor 192.168.3.1 send-community extended
 exit-address-family
!
address-family ipv4 vrf vrf_user
 redistribute connected
 neighbor 172.22.1.1 remote-as 40000
 neighbor 172.22.1.1 local-as 50000 no-prepend
 neighbor 172.22.1.1 ebgp-multihop 2
 neighbor 172.22.1.1 activate
 neighbor 172.22.1.1 allowas-in 1
 no auto-summary
 no synchronization
 bgp router-id auto-assign
 exit-address-family
!
address-family ipv4 vrf vrf_trans
 redistribute connected
 neighbor 172.23.1.1 remote-as 50000
 neighbor 172.23.1.1 local-as 40000 no-prepend
 neighbor 172.23.1.1 ebgp-multihop 2
 neighbor 172.23.1.1 activate
 neighbor 172.23.1.1 allowas-in 1
 no auto-summary
 no synchronization
 bgp router-id auto-assign
 exit-address-family

```

コンフィギュレーションの後、**show ip bgp vpnv4 all** コマンドの出力には、VRF 名の次に表示されるルータ ID が表示されます。この例で使用されているルータ ID が、ループバック インターフェイス 1 およびループバック インターフェイス 2 で設定された IP アドレスから送信されていることに注意してください。

```

Router# show ip bgp vpnv4 all
BGP table version is 43, local router ID is 10.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network        Next Hop        Metric LocPrf Weight Path
Route Distinguisher: 45000:1 (default for vrf vrf_trans) VRF Router ID 10.99.2.2
*> 172.22.0.0      0.0.0.0          0           32768 ?
r> 172.23.0.0      172.23.1.1       0           0 3 1 ?
*>i10.21.1.1/32    192.168.3.1      0          100      0 2 i
*> 10.52.1.0/24    172.23.1.1       0           0 3 1 ?
*> 10.52.2.1/32    172.23.1.1       0           0 3 1 3 i
*> 10.52.3.1/32    172.23.1.1       0           0 3 1 3 i
*> 10.99.1.1/32    172.23.1.1       0           0 3 1 ?

```

```

*> 10.99.1.2/32      0.0.0.0          0          32768 ?
Route Distinguisher: 50000:1
*>i10.21.1.1/32     192.168.3.1      0    100      0 2 i
Route Distinguisher: 65500:1 (default for vrf vrf_user) VRF Router ID 10.99.1.1
r> 172.22.0.0       172.22.1.1       0          0 2 1 ?
*> 172.23.0.0       0.0.0.0          0          32768 ?
*> 10.21.1.1/32     172.22.1.1       0          0 2 1 2 i
*>i10.52.1.0/24     192.168.3.1      0    100      0 ?
*>i10.52.2.1/32     192.168.3.1      0    100      0 3 i
*>i10.52.3.1/32     192.168.3.1      0    100      0 3 i
*> 10.99.1.1/32     0.0.0.0          0          32768 ?
*> 10.99.1.2/32     172.22.1.1       0          0 2 1 ?

```

その他の参考資料

関連資料

関連項目	マニュアルタイトル
BGP コマンド：コマンド構文の詳細、デフォルト、コマンドモード、コマンド履歴、使用上の注意事項、および例	『Cisco IOS IP Routing: BGP Command Reference』
MPLS コマンド：コマンド構文の詳細、デフォルト、コマンドモード、コマンド履歴、使用上の注意事項、および例	『Cisco IOS Multiprotocol Label Switching Command Reference』
『Cisco IOS Master Command List, All Releases』	『Cisco IOS Master Command List, All Releases』

標準

標準	タイトル
この機能でサポートされる新規の規格または変更された規格はありません。また、既存の規格のサポートは変更されていません。	--

MIB

MIB	MIB のリンク
この機能によってサポートされる新しい MIB または変更された MIB はありません。またこの機能による既存 MIB のサポートに変更はありません。	<p>選択したプラットフォーム、Cisco IOS XE ソフトウェアリリース、およびフィーチャセットの MIB の場所を検索しダウンロードするには、次の URL にある Cisco MIB Locator を使用します。</p> <p>http://www.cisco.com/go/mibs</p>

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	--

テクニカル サポート

説明	リンク
<p>シスコのサポート Web サイトでは、シスコの製品やテクノロジーに関するトラブルシューティングにお役立ていただけるように、マニュアルやツールをはじめとする豊富なオンラインリソースを提供しています。</p> <p>お使いの製品のセキュリティ情報や技術情報を入力するために、Cisco Notification Service (Field Notice からアクセス)、Cisco Technical Services Newsletter、Really Simple Syndication (RSS) フィードなどの各種サービスに加入できます。</p> <p>シスコのサポート Web サイトのツールにアクセスする際は、Cisco.com のユーザ ID およびパスワードが必要です。</p>	http://www.cisco.com/en/US/support/index.html

BGP ルータ ID の VRF 単位での割り当てに関する機能情報

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

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

表 1: BGP ルータ ID の VRF 単位での割り当てに関する機能情報

機能名	リリース	機能情報
BGP ルータ ID の VRF 単位での割り当て	Cisco IOS XE Release 2.1	<p>BGP ルータ ID の VRF 単位の割り当て機能により、同じルータ上のボーダーゲートウェイプロトコル (BGP) 内に VRF-to-VRF ピアリングを持つ機能が追加されます。BGP は、ルータ ID チェックのため、BGP 自身でセッションを拒否するように設計されています。VRF 単位の割り当て機能を使用すると、既存の bgp router-id コマンドの新しいキーワードを使用して、VRF 単位で異なるルータ ID を使用できます。ルータ ID は、VRF 単位での手動設定、または、アドレス ファミリ コンフィギュレーション モードでのグローバルな自動割り当てや VRF 単位の自動割り当てが可能です。</p> <p>この機能は、Cisco ASR 1000 シリーズのアグリゲーション サービス ルータで導入されました。</p> <p>この機能により、次のコマンドが導入または変更されました。 bgp router-id、show ip bgp vpv4</p>

