Configuring BGP: RT Constrained Route Distribution

BGP: RT Constrained Route Distribution is a feature that can be used by service providers in Multiprotocol Label Switching (MPLS) Layer 3 VPNs to reduce the number of unnecessary routing updates that route reflectors (RRs) send to Provider Edge (PE) routers. The reduction in routing updates saves resources by allowing RRs, Autonomous System Boundary Routers (ASBRs), and PEs to have fewer routes to carry. Route targets are used to constrain routing updates.

- Finding Feature Information, page 1
- Prerequisites for BGP: RT Constrained Route Distribution, page 2
- Restrictions for BGP: RT Constrained Route Distribution, page 2
- Information About BGP: RT Constrained Route Distribution, page 2
- How to Configure RT Constrained Route Distribution, page 6
- Configuration Examples for BGP: RT Constrained Route Distribution, page 17
- Additional References, page 18
- Feature Information for BGP: RT Constrained Route Distribution, page 20

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for BGP: RT Constrained Route Distribution

Before you configure BGP: RT Constrained Route Distribution, you should understand how to configure the following:

- Multiprotocol Label Switching (MPLS) VPNs
- Route distinguishers (RDs)
- Route targets (RTs)
- Multiprotocol BGP (MBGP)

Restrictions for BGP: RT Constrained Route Distribution

BGP: RT Constrained Route Distribution constrains all VPN route advertisements.

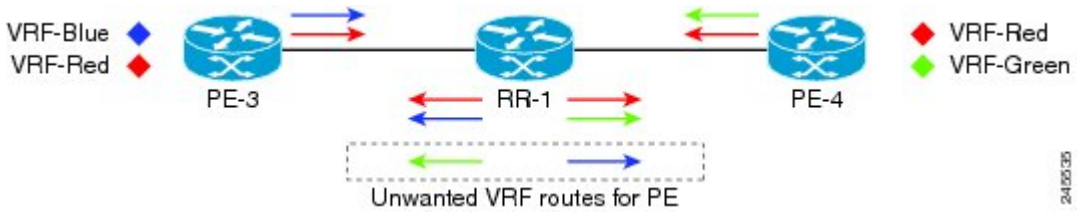
Information About BGP: RT Constrained Route Distribution

Problem That BGP: RT Constrained Route Distribution Solves

Some service providers have a large number of routing updates being sent from RRs to PEs, which can require extensive use of resources. A PE does not need routing updates for VRFs that are not on the PE; therefore, the PE determines that many routing updates it receives are "unwanted." The PE filters out the unwanted updates.

The figure below illustrates a scenario in which unwanted routing updates arrive at two PEs.

Figure 1: Unwanted Routing Updates at PE

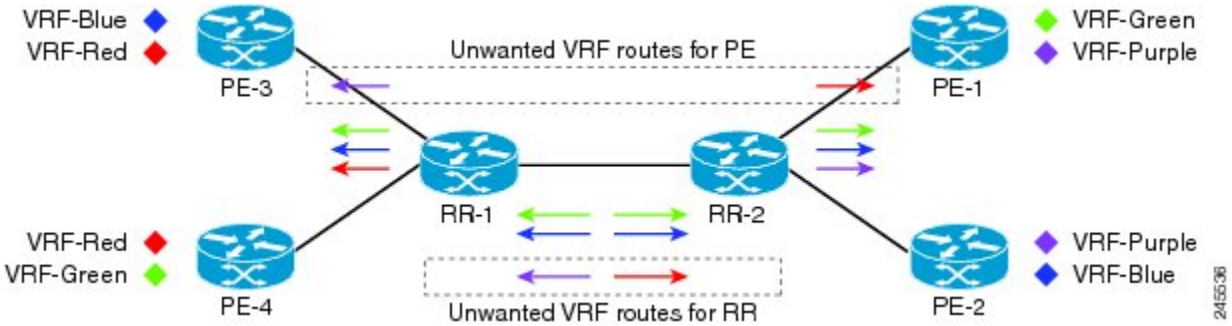


As shown in the figure above, a PE receives unwanted routes in the following manner:

1. PE-3 advertises VRF Blue and VRF Red routes to RR-1. PE-4 advertises VRF Red and VRF Green routes to RR-1.
2. RR-1 has all of the routes for all of the VRFs (Blue, Red, and Green).
3. During a route refresh or VRF provisioning, RR-1 advertises all of the VRF routes to both PE-3 and PE-4.
4. Routes for VRF Green are unwanted at PE-3. Routes for VRF Blue are unwanted at PE-4.

Now consider the scenario where there are two RRs with another set of PEs. There are unwanted routing updates from RRs to PEs and unwanted routing updates between RRs. The figure below illustrates a scenario in which unwanted routes arrive at an RR.

Figure 2: Unwanted Routing Updates at RR



As shown in the figure above, RR-1 and RR-2 receive unwanted routing updates in the following manner:

1. PE-3 and PE-4 advertise VRF Blue, VRF Red, and VRF Green VPN routes to RR-1.
2. RR-1 sends all of its VPN routes to RR-2.
3. VRF Red routes are unwanted on RR-2 because PE-1 and PE-2 do not have VRF Red.
4. Similarly, VRF Purple routes are unwanted on RR-1 because PE-3 and PE-4 do not have VRF Purple.

Hence, a large number of unwanted routes might be advertised among RRs and PEs. The BGP: RT Constrained Route Distribution feature addresses this problem by filtering unwanted routing updates.

Before the BGP: RT Constrained Route Distribution feature, the PE would filter the updates. With this feature, the burden is moved to the RR to filter the updates.

Benefits of BGP: RT Constrained Route Distribution

In MPLS L3VPNs, PE routers use BGP and route target (RT) extended communities to control the distribution of VPN routes to and from VRFs in order to separate the VPNs. PEs and Autonomous System Boundary Routers (ASBRs) commonly receive and then filter out the unwanted VPN routes.

However, receiving and filtering unwanted VPN routes is a waste of resources. The sender generates and transmits a VPN routing update and the receiver filters out the unwanted routes. Preventing the generation of VPN route updates would save resources.

Route Target Constrain (RTC) is a mechanism that prevents the propagation of VPN Network Layer Reachability Information (NLRI) from the RR to a PE that is not interested in the VPN. The feature provides considerable savings in CPU cycles and transient memory usage. RT constraint limits the number of VPN routes and describes VPN membership.

BGP RT-Constrain SAFI

The BGP: RT Constrained Route Distribution feature introduces the BGP RT-Constrain Subsequent Address Family Identifier (SAFI). The command to enter that address family is the **address-family rtfilter unicast** command.

BGP: RT Constrained Route Distribution Operation

In order to filter out the unwanted routes described in the "Problem that BGP RT Constrained Route Distribution Solves" section on page 2, the PEs and RRs must be configured with the BGP: RT Constrained Route Distribution feature.

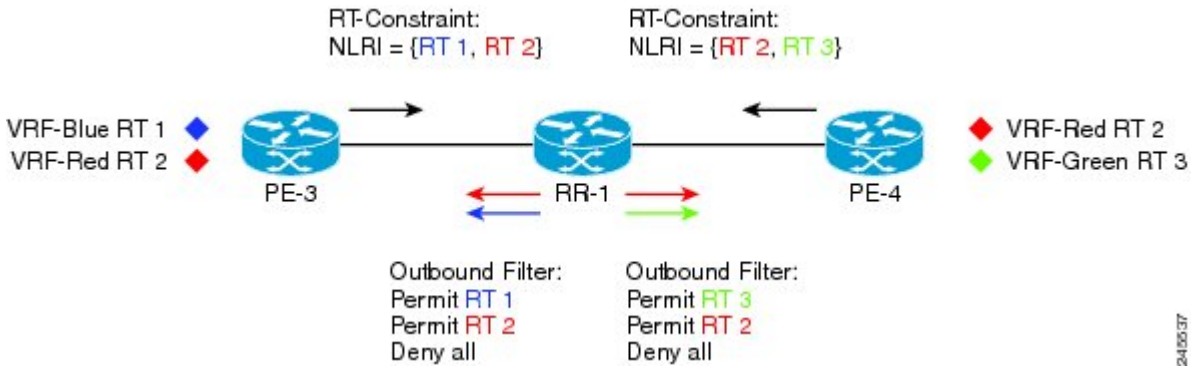
The feature allows the PE to propagate RT membership and use the RT membership to limit the VPN routing information maintained at the PE and RR. The PE uses an MP-BGP UPDATE message to propagate the membership information. The RR restricts advertisement of VPN routes based on the RT membership information it received.

This feature causes two exchanges to happen:

- The PE sends RT Constraint (RTC) Network Layer Reachability Information (NLRI) to the RR.
- The RR installs an outbound route filter.

The figure below illustrates the exchange of the RTC NLRI and the outbound route filter.

Figure 3: Exchange of RTC NLRI and Filter Between PE and RR



As shown in the figure above, the following exchange occurs between the PE and the RR:

1. PE-3 sends RTC NLRI (RT 1, RT 2) to RR-1.
2. PE-4 sends RTC NLRI (RT 2, RT 3) to RR-1.
3. RR-1 translates the NLRI into an outbound route filter and installs this filter (Permit RT 1, RT 2) for PE-3.
4. RR-1 translates the NLRI into an outbound route filter and installs this filter (Permit RT 2, RT 3) for PE-4.

RT Constraint NLRI Prefix

The format of the RT Constraint NLRI is a prefix that is always 12 bytes long, consisting of the following:

- 4-byte origin autonomous system
- 8-byte RT extended community value

The following are examples of RT Constraint prefixes:

- 65000:2:100:1
  - Origin autonomous system number is 65000
  - BGP Extended Community Type Code is 2
  - Route target is 100:1
- 65001:256:192.0.0.1:100
  - Origin ASN is 65001
  - BGP Extended Community Type Code is 256
  - Route target is 192.0.0.1:100
- 1.10:512:1.10:2
  - Origin ASN is 4-byte, unique 1.10
  - BGP Extended Community Type Code is 512
  - Route target is 1.10:2

To determine what the BGP Extended Community Type Code means, refer to RFC 4360, *BGP Extended Communities Attribute*. In the first example shown, a 2 translates in hexadecimal to 0x002. In RFC 4360, 0x002 indicates that the value that follows the type code will be a two-octet AS specific route target.

RT Constrained Route Distribution Process

This section shows the RT Constrained Route Distribution process. In this example has two CE routers in AS 100 that are connected to PE1. PE1 communicates with PE2, which is also connected to CE routers. Between the two PEs is a route reflector (RR). PE1 and PE2 belong to AS 65000.

The general process for the feature is as follows:

1. The user configures PE1 to activate its BGP peers under the **address-family rtfilter unicast** command.
2. The user configures PE1 in AS 65000 with **route-target import 100:1**, for example.
3. PE1 translates that command to an RT prefix of 65000:2:100:1. The 65000 is the service provider's AS number; the 2 is the BGP Extended Communities Type Code; and the 100:1 is the CE's RT (AS number and another number).
4. PE1 advertises the RT Constrain (RTC) prefix of 65000:2:100:1 to its iBGP peer RR.

5 The RR installs RTC 65000:2:100:1 into the RTC RIB. Each VRF has its own RIB.

6 The RR also installs RTC 65000:2:100:1 into its outbound filter for the neighbor PE1.

7 A filter in the RR either permits or denies the RT. (The AS number is ignored because iBGP is operating in a single AS and does not need to track the AS number.)

8 The RR looks in its outbound filter and sees that it permits outbound VPN packets for RT 100:1 to PE1. So, the RR sends VPN update packet only with RT 100:1 to PE1 and denies VPN updates with any other RT.

Default RT Filter

The default RT filter has a value of zero and length of zero. The default RT filter is used:

- By a peer to indicate that the peer wants all of the VPN routes sent to it, regardless of the RT value.
- By the RR to request that the PE advertise all of its VPN routes to the RR.

The default RT filter is created by configuring the **neighbor default-originate** command under the **address-family rtfilter unicast** command. On the RR it comes as default along with the configuration of route-reflector-client under the address-family rtfilter.

How to Configure RT Constrained Route Distribution

Configuring Multiprotocol BGP on Provider Edge (PE) Routers and Route Reflectors

Perform this task to configure multiprotocol BGP (MP-BGP) connectivity on the PE routers and route reflectors.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router bgp** *as-number*
4. **no bgp default ipv4-unicast**
5. **neighbor** {*ip-address* | *peer-group-name*} **remote-as** *as-number*
6. **address-family vpnv4** [**unicast**]
7. **neighbor** {*ip-address* | *peer-group-name*} **send-community extended**
8. **neighbor** {*ip-address* | *peer-group-name*} **activate**
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**router bgp** *as-number* **Example:** `Device(config)# router bgp 100`	Configures a BGP routing process and enters router configuration mode. • The *as-number* argument indicates the number of an autonomous system that identifies the router to other BGP routers and tags the routing information passed along. Valid numbers are from 0 to 65535. Private autonomous system numbers that can be used in internal networks range from 64512 to 65535.
Step 4	**no bgp default ipv4-unicast** **Example:** `Device(config-router)# no bgp default ipv4-unicast`	(Optional) Disables the IPv4 unicast address family on all neighbors. • Use the **no** form of the **bgp default ipv4-unicast** command if you are using this neighbor for MPLS routes only.
Step 5	**neighbor** {*ip-address* | *peer-group-name*} **remote-as** *as-number* **Example:** `Device(config-router)# neighbor pp.0.0.1 remote-as 100`	Adds an entry to the BGP or multiprotocol BGP neighbor table. • The *ip-address* argument specifies the IP address of the neighbor. • The *peer-group-name* argument specifies the name of a BGP peer group. • The *as-number* argument specifies the autonomous system to which the neighbor belongs.
Step 6	**address-family vpnv4** [**unicast**] **Example:** `Device(config-router)# address-family vpnv4`	Enters address family configuration mode for configuring routing sessions, such as BGP, that use standard VPNv4 address prefixes. • The optional **unicast** keyword specifies VPNv4 unicast address prefixes.
Step 7	**neighbor** {*ip-address* | *peer-group-name*} **send-community extended** **Example:** `Device(config-router-af)# neighbor pp.0.0.1 send-community extended`	Specifies that a communities attribute should be sent to a BGP neighbor. • The *ip-address* argument specifies the IP address of the BGP-speaking neighbor. • The *peer-group-name* argument specifies the name of a BGP peer group.

	Command or Action	Purpose
Step 8	**neighbor** {*ip-address* | *peer-group-name*} **activate** **Example:** `Device(config-router-af)# neighbor pp.0.0.1 activate`	Enables the exchange of information with a neighboring BGP router. • The *ip-address* argument specifies the IP address of the neighbor. • The *peer-group-name* argument specifies the name of a BGP peer group.
Step 9	**end** **Example:** `Device(config-router-af)# end`	(Optional) Exits to privileged EXEC mode.

Troubleshooting Tips

You can enter a **show ip bgp neighbor** command to verify that the neighbors are up and running. If this command is not successful, enter a **debug ip bgp** *ip-address* **events** command, where *ip-address* is the IP address of the neighbor.

Connecting the MPLS VPN Customers

To connect the MPLS VPN customers to the VPN, perform the following tasks:

Defining VRFs on PE Routers to Enable Customer Connectivity

To define virtual routing and forwarding (VRF) instances, perform this task.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip vrf** *vrf-name*
4. **rd** *route-distinguisher*
5. **route-target** {**import** | **export** | **both**} *route-target-ext-community*
6. **import map** *route-map*
7. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ip vrf** *vrf-name* **Example:** `Device(config)# ip vrf vpn1`	Defines the VPN routing instance by assigning a VRF name and enters VRF configuration mode. • The *vrf-name* argument is the name assigned to a VRF.

Step 4	**rd** *route-distinguisher*
	
	Example:
	
	`Device(config-vrf)# rd 100:1`
		Creates routing and forwarding tables.
		
		• The *route-distinguisher* argument adds an 8-byte value to an IPv4 prefix to create a VPN IPv4 prefix. You can enter an RD in either of these formats:
		
		  • 16-bit AS number: your 32-bit number, for example, 101:3
		
		  • 32-bit IP address: your 16-bit number, for example, 192.168.122.15:1

Step 5	**route-target** {**import** | **export** | **both**} *route-target-ext-community* **Example:** `Device(config-vrf)# route-target import 100:1`	Creates a route-target extended community for a VRF. • The **import** keyword imports routing information from the target VPN extended community. • The **export** keyword exports routing information to the target VPN extended community. • The **both** keyword imports routing information from and exports routing information to the target VPN extended community. • The *route-target-ext-community* argument adds the RT extended community attributes to the VRF's list of import, export, or both (import and export) RT extended communities.
Step 6	**import map** *route-map* **Example:** `Device(config-vrf)# import map vpn1-route-map`	(Optional) Configures an import route map for a VRF. • The *route-map* argument specifies the route map to be used as an import route map for the VRF.

	Command or Action	Purpose
Step 7	**exit** **Example:** `Device(config-vrf)# exit`	(Optional) Exits to global configuration mode.

Configuring VRF Interfaces on PE Routers for Each VPN Customer

To associate a VRF with an interface or subinterface on the PE routers, perform this task.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip vrf forwarding** *vrf-name*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Device(config)# interface Ethernet 5/0`	Specifies the interface to configure and enters interface configuration mode. • The *type* argument specifies the type of interface to be configured. • The *number* argument specifies the port, connector, or interface card number.
Step 4	**ip vrf forwarding** *vrf-name*	Associates a VRF with the specified interface or subinterface.

	Command or Action	Purpose
	Example: `Device(config-if)# ip vrf forwarding vpn1`	• The *vrf-name* argument is the name assigned to a VRF.
Step 5	**end** **Example:** `Device(config-if)# end`	(Optional) Exits to privileged EXEC mode.

Configuring BGP as the Routing Protocol Between the PE and CE Routers

To configure PE-to-CE routing sessions using BGP, perform this task.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router bgp** *as-number*
4. **address-family ipv4** [**multicast** | **unicast** | **vrf** *vrf-name*]
5. **neighbor** {*ip-address* | *peer-group-name*} **remote-as** *as-number*
6. **neighbor** {*ip-address* | *peer-group-name*} **activate**
7. **exit-address-family**
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**router bgp** *as-number*	Configures a BGP routing process and enters router configuration mode.

	Command or Action	Purpose
	Example: `Device(config)# router bgp 100`	• The *as-number* argument indicates the number of an autonomous system that identifies the router to other BGP routers and tags the routing information passed along. Valid numbers are from 0 to 65535. Private autonomous system numbers that can be used in internal networks range from 64512 to 65535.
Step 4	**address-family ipv4** [**multicast** | **unicast** | **vrf** *vrf-name*] **Example:** `Device(config-router)# address-family ipv4 vrf vpn1`	Specifies the IPv4 address family type and enters address family configuration mode. • The **multicast** keyword specifies IPv4 multicast address prefixes. • The **unicast** keyword specifies IPv4 unicast address prefixes. • The **vrf** *vrf-name* keyword and argument specify the name of the VRF to associate with subsequent IPv4 address family configuration mode commands.
Step 5	**neighbor** {*ip-address* | *peer-group-name*} **remote-as** *as-number* **Example:** `Device(config-router-af)# neighbor pp.0.0.1 remote-as 200`	Adds an entry to the BGP or multiprotocol BGP neighbor table. • The *ip-address* argument specifies the IP address of the neighbor. • The *peer-group-name* argument specifies the name of a BGP peer group. • The *as-number* argument specifies the autonomous system to which the neighbor belongs.
Step 6	**neighbor** {*ip-address* | *peer-group-name*} **activate** **Example:** `Device(config-router-af)# neighbor pp.0.0.1 activate`	Enables the exchange of information with a neighboring BGP router. • The *ip-address* argument specifies the IP address of the neighbor. • The *peer-group-name* argument specifies the name of a BGP peer group.
Step 7	**exit-address-family** **Example:** `Device(config-router-af)# exit-address-family`	Exits address family configuration mode.
Step 8	**end** **Example:** `Device(config-router)# end`	(Optional) Exits to privileged EXEC mode.

Configuring RT Constraint on the PE

Perform this task on the PE to configure BGP: RT Constrained Route Distribution with the specified neighbor, and optionally verify that route target (RT) filtering is occurring.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router bgp** *as-number*
4. **address-family rtfilter unicast**
5. **neighbor** {*ip-address* | *peer-group-name*} **activate**
6. **neighbor** {*ip-address* | *peer-group-name*} **send-community extended**
7. **end**
8. **show ip bgp rtfilter all**
9. **show ip bgp rtfilter all summary**
10. **show ip bgp vpnv4 all**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**router bgp** *as-number* **Example:** `Device(config)# router bgp 1`	Configures a BGP routing process and enters router configuration mode.
Step 4	**address-family rtfilter unicast** **Example:** `Device(config-router)# address-family rtfilter unicast`	Specifies the RT filter address family type and enters address family configuration mode.

	Command or Action	Purpose
Step 5	**neighbor** {*ip-address* | *peer-group-name*} **activate** **Example:** `Device(config-router-af)# neighbor 10.0.0.1 activate`	Enables the exchange of automated RT filter information with the specified BGP neighbor.
Step 6	**neighbor** {*ip-address* | *peer-group-name*} **send-community extended** **Example:** `Device(config-router-af)# neighbor pp.0.0.1 send-community extended`	Specifies that a communities attribute should be sent to a BGP neighbor. • The *ip-address* argument specifies the IP address of the BGP-speaking neighbor. • The *peer-group-name* argument specifies the name of a BGP peer group.
Step 7	**end** **Example:** `Device(config-router-af)# end`	Exits configuration mode and returns to privileged EXEC mode.
Step 8	**show ip bgp rtfilter all** **Example:** `Device# show ip bgp rtfilter all`	(Optional) Displays all BGP RT filter information.
Step 9	**show ip bgp rtfilter all summary** **Example:** `Device# show ip bgp rtfilter all summary`	(Optional) Displays summary BGP RT filter information.
Step 10	**show ip bgp vpnv4 all** **Example:** `Device# show ip bgp vpnv4 all`	(Optional) Displays summary BGP VPNv4 information.

Configuring RT Constraint on the RR

Perform this task on the RR to configure BGP: RT Constrained Route Distribution with the specified neighbor, and optionally verify that route target (RT) filtering is occurring.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router bgp** *as-number*
4. **address-family rtfilter unicast**
5. **neighbor** {*ip-address* | *peer-group-name*} **activate**
6. **neighbor** {*ip-address* | *peer-group-name*} **route-reflector-client**
7. **neighbor** {*ip-address* | *peer-group-name*} **send-community extended**
8. **end**
9. **show ip bgp rtfilter all**
10. **show ip bgp rtfilter all summary**
11. **show ip bgp vpnv4 all**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**router bgp** *as-number* **Example:** `Device(config)# router bgp 1`	Configures a BGP routing process and enters router configuration mode.
Step 4	**address-family rtfilter unicast** **Example:** `Device(config-router)# address-family rtfilter unicast`	Specifies the RT filter address family type and enters address family configuration mode.
Step 5	**neighbor** {*ip-address* | *peer-group-name*} **activate** **Example:** `Device(config-router-af)# neighbor 10.0.0.2 activate`	Enables RT Constraint with the specified BGP neighbor.

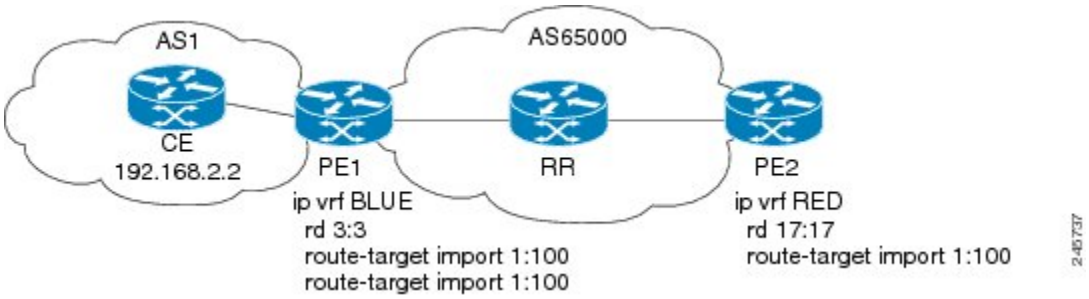
	Command or Action	Purpose
Step 6	**neighbor** {*ip-address* | *peer-group-name*} **route-reflector-client** **Example:** `Device(config-router-af)# neighbor 10.0.0.2 route-reflector-client`	Enables route-reflector-client funtionality under RT Constraint with the specified BGP neighbor. • Note that the route-reflector-client under RT Constraint address-family comes with a default "neighbor 10.0.0.2 default-originate" functionality that automatically gets added to the BGP configuration. The reason to have this is to have the route-reflector get all the VPN prefixes from its peer.
Step 7	**neighbor** {*ip-address* | *peer-group-name*} **send-community extended** **Example:** `Device(config-router-af)# neighbor 10.0.0.2 send-community extended`	Specifies that a communities attribute should be sent to a BGP neighbor. • The *ip-address* argument specifies the IP address of the BGP-speaking neighbor. • The *peer-group-name* argument specifies the name of a BGP peer group.
Step 8	**end** **Example:** `Device(config-router-af)# end`	Exits address family configuration mode and returns to privileged EXEC mode.
Step 9	**show ip bgp rtfilter all** **Example:** `Device# show ip bgp rtfilter all`	(Optional) Displays all BGP RT filter information.
Step 10	**show ip bgp rtfilter all summary** **Example:** `Device# show ip bgp rtfilter all summary`	(Optional) Displays summary BGP RT filter information.
Step 11	**show ip bgp vpnv4 all** **Example:** `Device# show ip bgp vpnv4 all`	(Optional) Displays summary BGP VPNv4 information.

Configuration Examples for BGP: RT Constrained Route Distribution

Example: BGP RT Constrained Route Distribution Between a PE and RR

The following example provides the configurations of the routers in the figure below. PE1 and PE2 are each connected to the RR and belong to AS 65000.

Figure 4: BGP: RT Constrained Route Distribution Between a PE and RR



PE1 Configuration

```
ip vrf BLUE
 rd 3:3
 route-target export 1:100
 route-target import 1:100
!
router bgp 65000
 bgp log-neighbor-changes
 neighbor 192.168.2.2 remote-as 65000
 neighbor 192.168.2.2 update-source Loopback0
 no auto-summary
 !
 address-family vpnv4
  neighbor 192.168.2.2 activate
  neighbor 192.168.2.2 send-community extended
 exit-address-family
 !
 address-family rtfilter unicast
  neighbor 192.168.2.2 activate
  neighbor 192.168.2.2 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf BLUE
  redistribute static
 exit-address-family
 !
ip route vrf BLUE 51.51.51.51 255.255.255.255 Null0
!
```

RR Configuration

```
!
router bgp 65000
 bgp log-neighbor-changes
```

```
 neighbor 192.168.6.6 remote-as 65000
 neighbor 192.168.6.6 update-source Loopback0
 neighbor 192.168.7.7 remote-as 65000
 neighbor 192.168.7.7 update-source Loopback0
 !
 address-family vpnv4
  neighbor 192.168.6.6 activate
  neighbor 192.168.6.6 send-community extended
  neighbor 192.168.6.6 route-reflector-client
  neighbor 192.168.7.7 activate
  neighbor 192.168.7.7 send-community extended
  neighbor 192.168.7.7 route-reflector-client
 exit-address-family
 !
 address-family rtfilter unicast
  neighbor 192.168.6.6 activate
  neighbor 192.168.6.6 send-community extended
  neighbor 192.168.6.6 route-reflector-client
  neighbor 192.168.6.6 default-originate
  neighbor 192.168.7.7 activate
  neighbor 192.168.7.7 send-community extended
  neighbor 192.168.7.7 route-reflector-client
  neighbor 192.168.7.7 default-originate
 exit-address-family
!
```

PE2 Configuration

```
!
ip vrf RED
 rd 17:17
 route-target export 150:15
 route-target import 150:1
 route-target import 1:100
!
router bgp 65000
 bgp log-neighbor-changes
 neighbor 192.168.2.2 remote-as 65000
 neighbor 192.168.2.2 update-source Loopback0
 neighbor 192.168.2.2 weight 333
 no auto-summary
 !
 address-family vpnv4
  neighbor 192.168.2.2 activate
  neighbor 192.168.2.2 send-community extended
exit-address-family
 !
 address-family rtfilter unicast
  neighbor 192.168.2.2 activate
  neighbor 192.168.2.2 send-community extended
  exit-address-family
!
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
BGP commands	Cisco IOS IP Routing: BGP Command Reference

Related Topic	Document Title
L3VPNs and route targets	"Configuring MPLS Layer 3 VPNs" module in the *MPLS: Layer 3 VPNs Configuration Guide*
MPLS commands	Cisco IOS Multiprotocol Label Switching Command Reference

MIBs

MIB	MIBs Link
—	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 4360	*BGP Extended Communities Attribute*
RFC 4684	*Constrained Route Distribution for Border Gateway Protocol/MultiProtocol Label Switching (BGP/MPLS) Internet Protocol (IP) Virtual Private Networks (VPNs)*
RFC 5291	*Outbound Route Filtering Capability for BGP-4*

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for BGP: RT Constrained Route Distribution

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 1: Feature Information for BGP: RT Constrained Route Distribution

Feature Name	Releases	Feature Information
BGP: RT Constrained Route Distribution	15.1(1)S 15.2(3)T 15.2(4)S 15.1(1)SY	BGP: Route Target (RT) Constrained Route Distribution is a feature that service providers can use in MPLS L3VPNs to reduce the number of unnecessary routes that RRs send to PEs, and thereby save resources. The following commands were introduced: **address-family rtfilter unicast** and **show ip bgp rtfilter** . In Cisco IOS Release 15.2(4)S, support was added for the Cisco 7200 series router.