



BGP-VRF-Aware Conditional Advertisement

The Border Gateway Protocol (BGP) VRF-Aware Conditional Advertisement feature provides additional control of the advertisement of routes and extends this control to within a virtual routing and forwarding (VRF) instance.

- [Information About BGP VRF-Aware Conditional Advertisement, on page 1](#)
- [How to Configure BGP VRF-Aware Conditional Advertisement, on page 2](#)
- [Configuration Examples for BGP VRF-Aware Conditional Advertisement, on page 5](#)
- [Additional References for BGP VRF-Aware Conditional Advertisement , on page 9](#)
- [Feature Information for BGP VRF-Aware Conditional Advertisement, on page 10](#)

Information About BGP VRF-Aware Conditional Advertisement

VRF-Aware Conditional Advertisement

The Border Gateway Protocol (BGP) VRF-Aware Conditional Advertisement feature provides additional control of the advertisement of routes and extends this control within a virtual routing and forwarding (VRF) instance.

BGP Conditional Advertisement

Normally, routes are propagated regardless of the existence of a different route. The BGP conditional advertisement feature uses the **exist-map**, **non-exist-map**, and the **advertise-map** keywords of the **neighbor** command in order to track routes by the route prefix. If a route prefix is not present in output of the **non-exist-map** command, then the route specified by the **advertise-map** is announced. This feature is useful for multihomed networks, in which some prefixes are advertised to one of the providers only if information from the other provider is not present (this indicates a failure in the peering session or partial reachability). The conditional BGP announcements are sent in addition to the normal announcements that a BGP router sends to its peers.

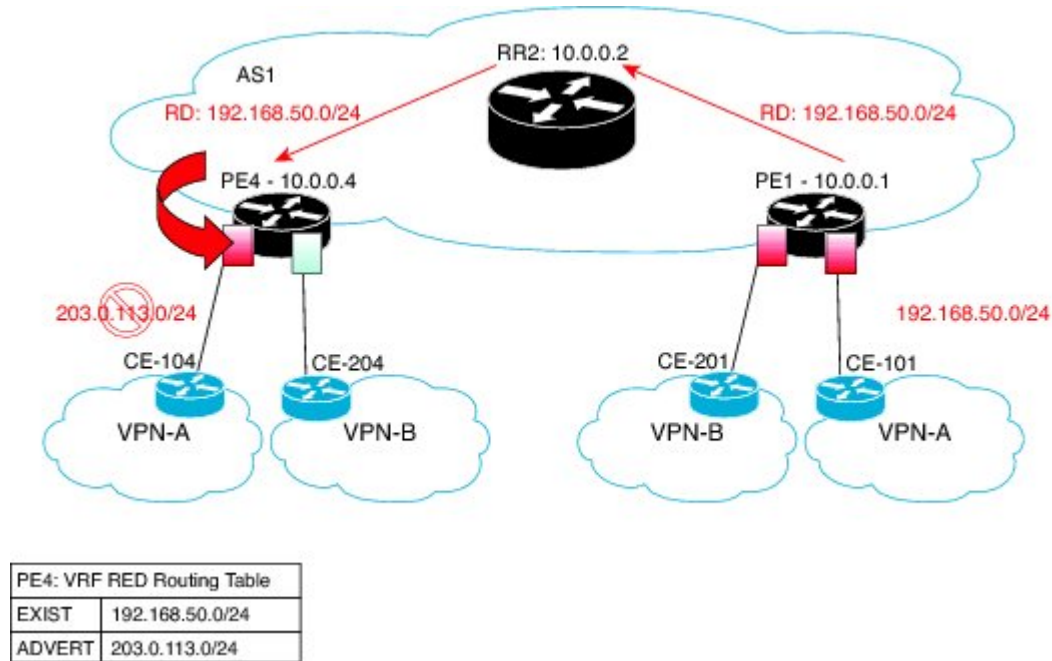
VRF-Aware Conditional Advertisement

This feature extends support for BGP VRF-aware conditional advertisement to the following address families:

- IPv4 unicast
- IPv4 unicast VRF

- IPv6 unicast
- IPv6 unicast VRF

Figure 1: VRF-Based Conditional Advertisement



The figure above shows the IPv4 prefix 192.168.50.0/24 being advertised by a remote CE101 into VRF RED on PE1. The prefix flows as a MP-BGP VPN prefix and is imported into the VRF RED on PE4. On the PE4 the conditions configured by the **exist-map** command relating to this prefix in the BGP VRF RED table becomes the condition to advertise the prefix 203.0.113.0/24 to the CE104, that is, peer-activated under the VRF RED on the PE4. This scenario assumes that 203.0.113.0/24 is in the VRF RED BGP table. If 203.0.113.0/24 is not in the table, this policy is ignored.

- If 192.168.50.0/24 exists in PE4's BGP table, then the 203.0.113.0/24 network is advertised to CE104.
- If 192.168.50.0/24 does not exist in PE4's BGP table, then the 203.0.113.0/24 network is not advertised to CE104.

How to Configure BGP VRF-Aware Conditional Advertisement

Configuring BGP VRF-Aware Conditional Advertisement

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router bgp** *autonomous-system-number*
4. Enter one of the following:

- **address-family ipv4** [unicast] [vrf vrf-name]

- **address-family ipv6** [unicast] [vrf vrf-name]

5. **neighbor** {ip-address | ipv6-address} **remote-as** autonomous-system-number

6. **neighbor** {ip-address | ipv6-address} **activate**

7. **neighbor** {ip-address | ipv6-address} **advertise-map** map-name {**exist-map** map-name | **non-exist-map** map-name}

8. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	router bgp autonomous-system-number Example: <pre>Device(config)# router bgp 40000</pre>	Enters router configuration mode for the specified routing process.
Step 4	Enter one of the following: • address-family ipv4 [unicast] [vrf vrf-name] • address-family ipv6 [unicast] [vrf vrf-name] Example: <pre>Device(config-router)# address-family ipv4 vrf VRFRED</pre>	Specifies the IPv4 or IPv6 address family and enters address family configuration mode. • The unicast keyword specifies the IPv4 or IPv6 unicast address family. • The vrf keyword and <i>vrf-name</i> argument specify the name of the virtual routing and forwarding (VRF) instance to associate with subsequent IPv4 or IPv6 address family configuration mode commands.
Step 5	neighbor {ip-address ipv6-address} remote-as autonomous-system-number Example: <pre>Device(config-router-af)# neighbor 192.0.2.1 remote-as 104</pre>	Adds the IP address of the neighbor in the specified autonomous system to the IPv4 or IPv6 multiprotocol BGP neighbor table of the local device.

	Command or Action	Purpose
Step 6	neighbor { <i>ip-address</i> <i>ipv6-address</i> } activate Example: <pre>Device(config-router-af)# neighbor 192.0.2.1 activate</pre>	Enables the neighbor to exchange prefixes for the IPv4 or IPv6 address family with the local device.
Step 7	neighbor { <i>ip-address</i> <i>ipv6-address</i> } advertise-map <i>map-name</i> { exist-map <i>map-name</i> non-exist-map <i>map-name</i> } Example: <pre>Device(config-router-af)# neighbor 192.0.2.1 advertise-map ADV-1 exist-map EXIST-1</pre>	Enables conditional advertisement towards a neighbor to allow the advertisement of prefixes mapped by the advertise-map command based on the criteria defined under exist or non-exist maps. • The advertise-map <i>map-name</i> keyword-argument pair specifies the name of the route map used to define the advertised routes. • The exist-map <i>map-name</i> keyword-argument pair specifies the condition that can be satisfied by a set of routes in the BGP table. If the condition is satisfied then the routes in the BGP table matching those specified in advertise map will be advertised. If the routes matching those specified in exist-map do not exist in the BGP table, those routes will not be advertised. • The non-exist-map <i>map-name</i> keyword-argument pair specifies the condition that is compared to a set of routes in the BGP table. If the routes in the non-exist-map are not present in the BGP table, then the routes matching those specified in advertise map will be advertised. If the routes matching those specified in non-exist-map are present in the BGP table, then the routes matching advertise-map will not be advertised.
Step 8	end Example: <pre>Device(config-router-af)# end</pre>	Exits address family configuration mode and enters privileged EXEC mode.

What to do next

To verify the configuration of the BGP VRF-Aware Conditional Advertisement feature, use the **show bgp ip neighbors** command.

Configuration Examples for BGP VRF-Aware Conditional Advertisement

Example: Configuring BGP VRF-Aware Conditional Advertisement

The following examples use the configuration in figure 1:

CE 101: The source of the prefixes

```
router bgp 101
  bgp log-neighbor-changes
  timers bgp 0 0
  neighbor 172.16.1.2 remote-as 65000
  !
  address-family ipv4
    network 21.21.21.0 mask 255.255.255.0
    network 22.22.22.22 mask 255.255.255.255
    network 31.0.0.0
    network 33.0.0.0
    network 44.0.0.0
    network 192.0.254 mask 255.255.255.0
    network 192.0.2.50
    neighbor 172.16.1.3 activate
  exit-address-family
```

PE 1

```
router bgp 65000
  bgp log-neighbor-changes
  no bgp default ipv4-unicast
  timers bgp 0 0
  neighbor 10.0.0.2 remote-as 65000
  neighbor 10.0.0.2 update-source Loopback0
  !
  address-family ipv4
  exit-address-family
  !
  address-family vpnv4
    neighbor 10.0.0.2 activate
    neighbor 10.0.0.2 send-community both
  exit-address-family
  !
  address-family ipv4 vrf blue
    neighbor 198.51.100.10 remote-as 201
    neighbor 198.51.100.10 activate
  exit-address-family
  !
  address-family ipv4 vrf red
    neighbor 172.16.1.2 remote-as 101
    neighbor 172.16.1.2 activate
  exit-address-family
```

PE 4

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

Example: Configuring BGP VRF-Aware Conditional Advertisement

```

no bgp default ipv4-unicast
timers bgp 0 0
neighbor 10.0.0.2 remote-as 65000
neighbor 10.0.0.2 update-source Loopback0
!
address-family ipv4
exit-address-family
!
address-family vpnv4
neighbor 10.0.0.2 activate
neighbor 10.0.0.2 send-community extended
exit-address-family
!
address-family ipv4 vrf blue
neighbor 198.51.100.12 remote-as 204
neighbor 198.51.100.12 activate
exit-address-family
!
address-family ipv4 vrf red
neighbor 198.51.100.3 remote-as 104
neighbor 198.51.100.3 activate
neighbor 198.51.100.3 advertise-map ADV-1 exist-map EXIST-1
neighbor 198.51.100.3 advertise-map ADV-2 exist-map EXIST-2
neighbor 198.51.100.3 advertise-map ADV-3 exist-map EXIST-3
neighbor 198.51.100.3 advertise-map ADV-4 exist-map EXIST-4
exit-address-family
!
ip prefix-list pl-adv-1 seq 5 permit 22.22.22.22/32
!
ip prefix-list pl-adv-2 seq 5 permit 44.0.0.0/8 □
!
ip prefix-list pl-adv-3 seq 5 permit 33.0.0.0/8
!
ip prefix-list pl-adv-4 seq 5 permit 128.16.16.0/24
!
ip prefix-list pl-exist-1 seq 5 permit 21.21.21.0/24
!
ip prefix-list pl-exist-2 seq 5 permit 41.0.0.0/8 □
!
ip prefix-list pl-exist-3 seq 5 permit 31.0.0.0/8
!
ip prefix-list pl-exist-4 seq 5 permit 192.168.50.0/24
!
route-map EXIST-4 permit 10
match ip address prefix-list pl-exist-4
!
route-map ADV-4 permit 10
match ip address prefix-list pl-adv-4
!
route-map EXIST-2 permit 10
match ip address prefix-list pl-exist-2
!
route-map ADV-2 permit 10
match ip address prefix-list pl-adv-2
!
route-map EXIST-3 permit 10
match ip address prefix-list pl-exist-3
!
route-map ADV-3 permit 10
match ip address prefix-list pl-adv-3
!
route-map EXIST-1 permit 10
match ip address prefix-list pl-exist-1
!

```

```
route-map ADV-1 permit 10
match ip address prefix-list pl-adv-1
```

Example: Verifying BGP VRF-Aware Conditional Advertisement

The following examples use the configuration in figure 1:

CE 101

```
CE101# show ip bgp all
```

```
For address family: IPv4 Unicast
BGP table version is 28, local router ID is 203.0.113.11
  Network          Next Hop           Metric LocPrf Weight Path
*> 21.21.21.0/24    0.0.0.0             0           32768 i
*> 22.22.22.22/32   0.0.0.0             0           32768 i
*> 31.0.0.0          0.0.0.0             0           32768 i
*> 33.0.0.0          0.0.0.0             0           32768 i
*> 44.0.0.0          0.0.0.0             0           32768 i
*> 192.0.2.254/24   0.0.0.0             0           32768 i
*> 192.0.2.50       0.0.0.0             0           32768 i
```

PE 1

```
PE1# show ip bgp all
```

```
For address family: IPv4 Unicast
```

```
For address family: VPNv4 Unicast
```

```
BGP table version is 46, local router ID is 10.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
               x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

```
  Network          Next Hop           Metric LocPrf Weight Path
Route Distinguisher: 1:1 (default for vrf red)
*> 21.21.21.0/24    172.16.1.2          0           0 101 i
*> 22.22.22.22/32   172.16.1.2          0           0 101 i
*> 31.0.0.0          172.16.1.2          0           0 101 i
*> 33.0.0.0          172.16.1.2          0           0 101 i
*> 44.0.0.0          172.16.1.2          0           0 101 i
*> 192.0.2.254/24   172.16.1.2          0           0 101 i
*> 192.0.2.50       172.16.1.2          0           0 101 i
```

PE 4



Note The status is Withdraw for the exist-map EXIST-2 because the condition for advertisement has not been met.

```
PE4# show ip bgp all
```

```
For address family: VPNv4 Unicast
```

Example: Verifying BGP VRF-Aware Conditional Advertisement

```
BGP table version is 82, local router ID is 10.0.0.4
```

Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 1:1 (default for vrf red)					
*>i 21.21.21.0/24	10.0.0.1	0	100	0	101 i
*>i 22.22.22.22/32	10.0.0.1	0	100	0	101 i
*>i 31.0.0.0	10.0.0.1	0	100	0	101 i
*>i 33.0.0.0	10.0.0.1	0	100	0	101 i
*>i 44.0.0.0	10.0.0.1	0	100	0	101 I <- missing 41.0.0.0/8
*>i 192.0.2.254/24	10.0.0.1	0	100	0	101 i
*>i 192.0.2.50	10.0.0.1	0	100	0	101 i

```
PE4# show ip bgp vpnv4 all neighbors 198.51.100.3
```

```
...
...
For address family: VPNv4 Unicast
Translates address family IPv4 Unicast for VRF red
Session: 198.51.100.3
BGP table version 48, neighbor version 48/0
Output queue size : 0
Index 3, Advertise bit 0
3 update-group member
Condition-map EXIST-1, Advertise-map ADV-1, status: Advertise
Condition-map EXIST-2, Advertise-map ADV-2, status: Withdraw
Condition-map EXIST-3, Advertise-map ADV-3, status: Advertise
Condition-map EXIST-4, Advertise-map ADV-4, status: Advertise
Slow-peer detection is disabled
...
...
PE4#
```

```
PE4# show ip bgp vpnv4 all update-group
```

```
...
...
BGP version 4 update-group 3, external, Address Family: VPNv4 Unicast
BGP Update version : 48/0, messages 0
Condition-map EXIST-1, Advertise-map ADV-1, status: Advertise
Condition-map EXIST-2, Advertise-map ADV-2, status: Withdraw
Condition-map EXIST-3, Advertise-map ADV-3, status: Advertise
Condition-map EXIST-4, Advertise-map ADV-4, status: Advertise
Topology: red, highest version: 47, tail marker: 47
Format state: Current working (OK, last not in list)
Refresh blocked (not in list, last not in list)
Update messages formatted 4, replicated 4, current 0, refresh 0, limit 1000
Number of NLRI in the update sent: max 3, min 0
Minimum time between advertisement runs is 0 seconds
Has 1 member:
198.51.100.3
```

CE 104



Note Prefix 44.0.0.0 is missing as 41.0.0.0/8 does not appear in PE 4 to trigger the advertisement to CE 104. The state is Withdraw.

```
CE104# show ip bgp all
```

For address family: IPv4 Unicast

BGP table version is 45, local router ID is 198.51.100.3

Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
x best-external, a additional-path, c RIB-compressed,

Origin codes: i - IGP, e - EGP, ? - incomplete

RPKI validation codes: V valid, I invalid, N Not found

	Network	Next Hop	Metric	LocPrf	Weight	Path
*>	21.21.21.0/24	104.0.0.1	0	65000	101	i
*>	22.22.22.22/32	104.0.0.1	0	65000	101	i
*>	31.0.0.0	104.0.0.1	0	65000	101	i
*>	33.0.0.0	104.0.0.1	0	65000	101	i
*>	192.0.2.254/24	104.0.0.1	0	65000	101	i
*>	192.0.2.50	104.0.0.1	0	65000	101	i

Additional References for BGP VRF-Aware Conditional Advertisement

Related Documents

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

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for BGP VRF-Aware Conditional Advertisement

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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1: Feature Information for BGP VRF-Aware Conditional Advertisement

Feature Name	Releases	Feature Information
BGP VRF-Aware Conditional Advertisement		The Border Gateway Protocol (BGP) VRF-Aware Conditional Advertisement feature provides additional control of the advertisement of routes and extends this control to within a virtual routing and forwarding (VRF) instance.