VRF-Aware IPv6 Rapid Deployment Tunnel

Virtual Routing and Forwarding - aware tunnels are used to connect customer networks separated by untrusted core networks or core networks with different infrastructures (IPv4 or IPv6). The VRF-Aware IPv6 Rapid Deployment Tunnel feature extends Virtual Routing and Forwarding (VRF) awareness to IPv6 rapid deployment tunnels.

- Finding Feature Information, on page 1
- Restrictions for the VRF-Aware IPv6 Rapid Deployment Tunnel, on page 1
- Information About the VRF-Aware IPv6 Rapid Deployment Tunnel, on page 2
- How to Configure the VRF-Aware IPv6 Rapid Deployment Tunnel, on page 2
- Feature Information for the VRF-Aware IPv6 Rapid Deployment Tunnel , on page 10

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.

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.

Restrictions for the VRF-Aware IPv6 Rapid Deployment Tunnel

The VRF- Aware IPv6 Rapid Deployment Tunnel feature has the following restrictions:

- The incoming physical interface, and the tunnel interface should have the same VRF instance defined.
- The tunnel transport VRF and the egress physical interface, through which the traffic leaves should have the same VRF instance defined.
- For IPv6 rapid deployment Customer Edge (CE) router configuration, the tunnel source and the Border Relay (BR) router address should have the same VRF instance defined as the physical interface through which the traffic flows.

Information About the VRF-Aware IPv6 Rapid Deployment Tunnel

The IPv6 Rapid Deployment Tunnel feature allows a service provider to provide a unicast IPv6 service to customers over its IPv4 network by using encapsulation of IPv6 in IPv4. Currently, the IPv6 Rapid Deployment Tunnel feature does not support VRF. Therefore, the forwarding table look up tasks for locating IPv6 overlay addresses and IPv4 transport addresses are performed in the global routing table. The VRF-Aware IPv6 Rapid Deployment Tunnel feature extends the IPv6 rapid deployment tunneling support for IPv6 overlay addresses and IPv4 transport addresses in VRF.

The following scenarios are supported for VRF-Aware IPv6 Rapid Deployment Tunnel feature:

- The IPv6 rapid deployment tunnel is in the VRF and both IPv6 overlay address and the IPv4 transport address are in VRF.
- IPv6 rapid deployment tunnel and the IPv4 address are in VRF. The incoming global routing table IPv6 traffic selects the correct VRF, based on the IPv6 rapid deployment domain.

The following figure explains the topology and sample configurations for the VRF Aware IPv6 Rapid Deployment Tunnel feature where both the IPv4 addresses and IPv6 addresses are in VRF.

Figure 1: Topology of the VRF-Aware IPv6 Rapid Deployment Tunnel

How to Configure the VRF-Aware IPv6 Rapid Deployment Tunnel

Complete the steps in the following procedure to configure the VRF-Aware IPv6 Rapid Deployment Tunnel feature when both the IPv6 and IPv4 addresses are in VRF. You should perform these steps on the CE router and BR router unless specifically mentioned otherwise in the following procedure.

Configuring the VRF-Aware IPv6 Rapid Deployment Tunnel

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 unicast-routing**
4. **vrf definition** *vrf-name1*
5. **rd** {*ASN:nn* | *IP address: nn*}
6. **route-target** [**import | export | both**] {*ASN:nn* |*IP address: nn*}
7. **address-family ipv6**
8. **exit**
9. **address-family ipv4**
10. **exit**
11. **exit**
12. **vrf definition** *vrf-name2*
13. **rd** {*ASN:nn* | *IP address: nn*}
14. **route-target** [**import | export | both**] { *ASN:nn* | *IP address: nn*}
15. **address-family ipv4**
16. **exit**

17. **exit**
18. **interface gigabitethernet** *slot / port*
19. **vrf forwarding** *vrf-name1*
20. **ipv6 address** {*ipv6-address prefix-length prefix-name sub-bits prefix-length*}
21. **exit**
22. **interface gigabitethernet** *slot* / *port*
23. **vrf forwarding** *vrf-name2*
24. **ip address** *ip-address mask*
25. **exit**
26. **interface loopback** *interface-number*
27. **vrf forwarding** *vrf-name2*
28. **ip address** *ip-address*
29. **exit**
30. **interface tunnel** *tunnel-number*
31. **vrf forwarding** *vrf-name1*
32. **ipv6 address** {*ipv6-address prefix-length prefix-name sub-bits prefix-length*}
33. **tunnel source**{ *ip-address*| *interface-type interface-number*}
34. **tunnel mode ipv6ip** [**6rd**| **6to4** | **auto-tunnel** | **isatap**]
35. **tunnel 6rd ipv4** {**prefix-length** *length*} {**suffix-length** *length*}
36. **tunnel 6rd prefix**{*ipv6-prefix*/ *prefix-length*}
37. **tunnel 6rd br** *ipv4-address*
38. **tunnel vrf** *vrf-name2*
39. **exit**
40. **ipv6 route vrf** *vrf-name1* {*ipv6-prefix* / *prefix-length*} **tunnel** *tunnel-number*
41. **ipv6 route vrf** *vrf-name1*{*ipv6-prefix*/ *prefix-length*} **tunnel** *tunnel-number ipv6-address*
42. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router# enable`	Enables the privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 3	**ipv6 unicast-routing** **Example:** `Router(config)# ipv6 unicast-routing`	Enables the task of forwarding IPv6 unicast datagrams.

	Command or Action	Purpose
Step 4	**vrf definition** *vrf-name1* **Example:** `Router(config)# vrf definition VRF_RED`	Configures a VRF instance and enters the VRF configuration mode.
Step 5	**rd** {*ASN:nn* | *IP address: nn*} **Example:** `Router(config-vrf)# rd 1:1`	Specifies a route distinguisher. • *ASN:nn* — Specifies an autonomous system number and an arbitrary number. • *IP address: nn* — Specifies an IP address and an arbitrary number.
Step 6	**route-target** [**import | export | both**] {*ASN:nn* |*IP address: nn*} **Example:** `Router(config-vrf)# route-target import 1:1`	Creates a route target extended community for a VRF instance. Route target extended community attributes are used to identify a set of sites and VRF instances that can receive routes with a configured route target. • **import** — Imports routing information from the target VPN extended community. • **export** — Exports routing information to the target VPN extended community. • **both** — Imports both import and export routing information to the target VPN extended community • *ASN:nn* — Specifies an autonomous system number and an arbitrary number. • *IP address: nn* — Specifies an IP address and an arbitrary number.
Step 7	**address-family ipv6** **Example:** `Router(config-vrf)# address-family ipv6`	Selects IPv6 as address family type for a VRF table and enters VRF address family configuration mode. Configures separate route-target policies for IPv6.
Step 8	**exit** **Example:** `Router(config-vrf-af)# exit`	Exits the address family configuration mode.
Step 9	**address-family ipv4** **Example:** `Router(config-vrf)# address-family ipv4`	Selects IPv4 as address family type for a VRF table and enters VRF address family configuration mode. Configures the separate route-target policies for IPv4.
Step 10	**exit** **Example:** `Router(config-vrf-af)# exit`	Exits the address family configuration mode.
Step 11	**exit**	Exits the VRF configuration mode.

	Command or Action	Purpose
	Example: `Router(config-vrf)# exit`	
Step 12	**vrf definition** *vrf-name2* **Example:** `Router(config)# vrf definition VRF_GREEN`	Configures a VRF instance and enters the VRF configuration mode.
Step 13	**rd** {*ASN:nn* | *IP address: nn*} **Example:** `Router(config-vrf)# rd 1:1`	Specifies a route distinguisher.
Step 14	**route-target** [**import | export | both**] { *ASN:nn* | *IP address: nn*} **Example:** `Router(config-vrf)# route-target import 1:1`	Creates a route-target extended community for a VRF instance. Route-target extended community attributes are used to identify a set of sites and VRF instances that can receive routes with a configured route target • **import** — Imports routing information from the target VPN extended community. • **export** — Exports routing information to the target VPN extended community. • **both** — Imports and exports routing information to the target VPN extended community and from the target VPN extended community. • *ASN:nn* — Specifies an autonomous system number and an arbitrary number. • *IP address: nn* — Specifies an IP address and an arbitrary number.
Step 15	**address-family ipv4** **Example:** `Router(config-vrf)# address-family ipv4`	Selects IPv4 as address family for a VRF table and enters the VRF address family configuration mode. Configures separate route-target policies for IPv4.
Step 16	**exit** **Example:** `Router(config-vrf-af)# exit`	Exits the address family configuration mode.
Step 17	**exit** **Example:** `Router(config-vrf)# exit`	Exits the VRF configuration mode.
Step 18	**interface gigabitethernet** *slot / port* **Example:** `Router(config)# interface gigabitethernet 3/1`	Enters the interface configuration mode and specifies the Gigabit interface to configure.

	Command or Action	Purpose
Step 19	**vrf forwarding** *vrf-name1* **Example:** `Router(config-if)# vrf forwarding VRF_RED`	Associates a VRF instance with an interface or a subinterface.
Step 20	**ipv6 address** {*ipv6-address prefix-length prefix-name sub-bits prefix-length*} **Example:** `Router(config-if)# ipv6 address 1::2/64`	Specifies the IPv6 address assigned to the interface, and enables IPv6 processing on the interface.
Step 21	**exit** **Example:** `Router(config-if)# exit`	Exits the interface configuration mode.
Step 22	**interface gigabitethernet** *slot* / *port* **Example:** `Router(config)# interface gigabitethernet 4/5`	Enters the interface configuration mode and specifies the Gigabit interface to configure.
Step 23	**vrf forwarding** *vrf-name2* **Example:** `Router(config-if)# vrf forwarding VRF_GREEN`	Associates a VRF instance with an interface or a subinterface.
Step 24	**ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 17.1.1.1 255.255.255.0`	Assigns an IP address and subnet mask to the interface.
Step 25	**exit** **Example:** `exit`	Exits the interface configuration mode.
Step 26	**interface loopback** *interface-number* **Example:** `Router(config)# interface Loopback 100`	Enters the interface configuration mode and specifies the new loopback interface.
Step 27	**vrf forwarding** *vrf-name2* **Example:** `Router(config-if)# vrf forwarding VRF_GREEN`	Associates a VRF instance with an interface or a subinterface.
Step 28	**ip address** *ip-address* **Example:** `Router(config-if)# ip address 60.1.1.1 255.255.255.0`	Assigns an IP address and subnet mask to the loopback interface.
Step 29	**exit**	Exits the interface configuration mode.

	Command or Action	Purpose
	Example: `Router(config-if)# exit`	
Step 30	**interface tunnel** *tunnel-number* **Example:** `Router(config)# interface tunnel 10`	Specifies a tunnel interface and enters the interface configuration mode.
Step 31	**vrf forwarding** *vrf-name1* **Example:** `Router(config-if)# vrf forwarding VRF_RED`	Associates a VRF instance with an interface or a subinterface.
Step 32	**ipv6 address** {*ipv6-address prefix-length prefix-name sub-bits prefix-length*} **Example:** `Router(config-if)# ipv6 address 2001:A000:100::1/128`	Specifies the IPv6 address assigned to the interface and enables IPv6 processing on the interface.
Step 33	**tunnel source**{ *ip-address*| *interface-type interface-number*} **Example:** `Router(config-if)# tunnel source loopback 100`	Specifies the source interface type and number for the tunnel interface.
Step 34	**tunnel mode ipv6ip** [**6rd**| **6to4** | **auto-tunnel** | **isatap**] **Example:** `Router(config-if)# tunnel mode ipv6ip 6rd`	Configures a static IPv6 tunnel interface.
Step 35	**tunnel 6rd ipv4** {**prefix-length** *length*} {**suffix-length** *length*} **Example:** `Router(config-if)# tunnel 6rd ipv4 prefix-len 16 suffix-len 8`	Specifies the prefix and suffix length of the IPv4 transport address that is common to all the 6rd tunnels.
Step 36	**tunnel 6rd prefix**{*ipv6-prefix*/ *prefix-length*} **Example:** `Router(config-if)# tunnel 6rd prefix 2001:A000::/32`	Specifies the common IPv6 prefix on IPv6 6rd tunnels.
Step 37	**tunnel 6rd br** *ipv4-address* **Example:** `Router(config-if)# tunnel 6rd br 60.1.2.1`	Bypasses security checks on a 6rd CE router. • *ipv4-address* — IPv4 address of the border relay (BR) router. **Note** Perform this step only on a CE router, not on a BR router.
Step 38	**tunnel vrf** *vrf-name2* **Example:**	Configures a VRF instance with a specific tunnel destination, interface, or a subinterface.

	Command or Action	Purpose
	`Router(config-if)# tunnel vrf  VRF_GREEN`	**Note** This command specifies the VRF instance used for the tunnel IPv4 transport address lookup.
Step 39	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode.
Step 40	**ipv6 route vrf** *vrf-name1* {*ipv6-prefix* / *prefix-length*} **tunnel** *tunnel-number* **Example:** `Router(config)# ipv6 route vrf VRF_RED 2001:A000::/32 Tunnel10`	Establishes static routes. • *ipv6-prefix* — Specifies the IPv6 network that is the destination of the static route. Can also be a host name when static host routes are configured. • *ipv6-address* — The IPv6 address of the next hop that can be used to reach the specified network..
Step 41	**ipv6 route vrf** *vrf-name1*{*ipv6-prefix*/ *prefix-length*} **tunnel** *tunnel-number ipv6-address* **Example:** `Router(config)# ipv6 route vrf VRF_RED 9000:1000::/64 Tunnel10 2001:A000:200::1`	Establishes static routes. • *ipv6-prefix* — Specifies the IPv6 network that is the destination of the static route. Can also be a host name when static host routes are configured. • *prefix-length* — Specifies the length of the IPv6 prefix.
Step 42	**end** **Example:** `Router(config)# end`	Ends the current configuration session.

Example: Configuring VRF- Aware IPv6 Rapid Deployment Tunnel

The following example shows how to configure the VRF-Aware IPv6 Rapid Deployment Tunnel on a CE router:

```
Router# enable
Router# configure terminal
Router(config)# ipv6 unicast-routing
Router(config)# mls ipv6 vrf
Router(config)# vrf definition VRF_RED
Router(config-vrf)# rd 1:1
Router(config-vrf)# route-target export 1:1
Router(config-vrf)# route-target import 1:1
Router(config-vrf)# address-family ipv6
Router(config-vrf-af)# exit
Router(config-vrf)# address-family ipv4
Router(config-vrf-af)# exit
Router(config-vrf)# exit
Router(config)# vrf definition VRF_GREEN
Router(config-vrf)# rd 1:1
Router(config-vrf)# route-target export 1:1
Router(config-vrf)# route-target import 1:1
```

```
Router(config-vrf)# address-family ipv4
Router(config-vrf-af)# exit
Router(config-vrf)# exit
Router(config)# interface gigabitethernet 3/1
Router(config-if)# vrf forwarding VRF_RED
Router(config-if)# ipv6 address 1::2/64
Router(config-if)# exit
Router(config)# interface gigabitethernet 4/5
Router(config-if)# vrf forwarding VRF_GREEN
Router(config-if)# ip address 17.1.1.1 255.255.255.0
Router(config-if)# ip ospf 2 area 0
Router(config-if)# exit
Router(config)# interface Loopback 100
Router(config-if)# vrf forwarding VRF_GREEN
Router(config-if)# ip address 60.1.1.1 255.255.255.0
Router(config-if)# ip ospf 2 area 0
Router(config-if)# exit
Router(config)# interface tunnel 10
Router(config-if)# vrf forwarding VRF_RED
Router(config-if)# ipv6 address 2001:A000:100::1/128
Router(config-if)# mls 6rd reserve interface GigabitEthernet4/5
Router(config-if)# tunnel source loopback 100
Router(config-if)# tunnel mode ipv6ip 6rd
Router(config-if)# tunnel 6rd ipv4 prefix-len 16 suffix-len 8
Router(config-if)# tunnel 6rd prefix 2001:A000::/32
Router(config-if)# tunnel 6rd br 60.1.2.1
Router(config-if)# tunnel vrf  VRF_GREEN
Router(config-if)# exit
Router(config)# ipv6 route vrf VRF_RED 2001:A000::/32 Tunnel10
Router(config)# ipv6 route vrf VRF_RED 9000:1000::/64 Tunnel10 2001:A000:200::1
Router(config)# end
```

The following example shows how to configure the VRF-Aware IPv6 Rapid Deployment Tunnel on a BR router:

```
Router# enable
Router# configure terminal
Router(config)# ipv6 unicast-routing
Router(config)# vrf definition VRF_RED
Router(config-vrf)# rd 1:1
Router(config-vrf)# route-target export 1:1
Router(config-vrf)# route-target import 1:1
Router(config-vrf)# address-family ipv6
Router(config-vrf-af)# exit
Router(config-vrf)# address-family ipv4
Router(config-vrf-af)# exit
Router(config-vrf)# exit
Router(config)# vrf definition VRF_GREEN
Router(config-vrf)# rd 1:1
Router(config-vrf)# route-target export 1:1
Router(config-vrf)# route-target import 1:1
Router(config-vrf)# address-family ipv4
Router(config-vrf-af)# exit
Router(config-vrf)# exit
Router(config)# interface gigabitethernet 5/1
Router(config-if)# vrf forwarding VRF_RED
Router(config-if)# ipv6 address 9000:1000::/64
Router(config-if)# exit
Router(config)# interface gigabitethernet 4/1
Router(config-if)# vrf forwarding VRF_GREEN
Router(config-if)# ip address 17.1.1.2 255.255.255.0
Router(config-if)# ip ospf 2 area 0
Router(config-if)# exit
Router(config)# interface Loopback 100
```

```
Router(config-if)# vrf forwarding VRF_GREEN
Router(config-if)# ip address 60.1.2.1 255.255.255.0
Router(config-if)# exit
Router(config)# interface tunnel 10
Router(config-if)# vrf forwarding VRF_RED
Router(config-if)# ipv6 address 2001:A000:100::1/128
Router(config-if)# tunnel source loopback 100
Router(config-if)# tunnel mode ipv6ip 6rd
Router(config-if)# tunnel 6rd ipv4 prefix-len 16 suffix-len 8
Router(config-if)# tunnel 6rd prefix 2001:A000::/32
Router(config-if)# tunnel vrf  VRF_GREEN
Router(config-if)# exit
Router(config)# ipv6 route vrf VRF_RED 2001:A000::/32 Tunnel10
Router(config)# end
```

Feature Information for the VRF-Aware IPv6 Rapid Deployment Tunnel

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 the VRF-Aware IPv6 Rapid Deployment Tunnel

Feature Name	Releases	Feature Information
VRF-Aware IPv6 Rapid Deployment Tunnel	Cisco IOS XE Release 3.10S	The IPv6 Rapid Deployment Tunnel feature allows a service provider to provide a unicast IPv6 service to customers over its IPv4 network by using encapsulation of IPv6 in IPv4. The VRF-Aware IPv6 Rapid Deployment Tunnel feature extends VRF awareness to IPv6 rapid deployment tunnels.