

Integrated Routing and Bridging

The BVI is a virtual interface within the router that acts like a normal routed interface. The BVI does not support bridging itself, but acts as a gateway for the corresponding bridge-domain to a routed interface within the router.

Aside from supporting a configurable MAC address, a BVI supports only Layer 3 attributes, and has the following characteristics:

- Uses a MAC address taken from the local chassis MAC address pool, unless overridden at the BVI interface.
- Is configured as an interface type using the **interface bvi** command and uses an IPv4 address that is in the same subnet as the hosts on the segments of the bridged domain.
- The BVI identifier is independent of the bridge-domain identifier. These identifiers do not need to correlate like they do in Cisco IOS software.
- Is associated to a bridge group using the **routed interface bvi** command.

- Supported Features on a BVI, on page 1
- BVI Interface and Line Protocol States, on page 2
- Prerequisites for Configuring IRB, on page 2
- Restrictions for Configuring IRB, on page 3
- How to Configure IRB, on page 3
- Additional Information on IRB, on page 9
- Packet Flows Using IRB, on page 9
- Configuration Examples for IRB, on page 11

Supported Features on a BVI

- These interface commands are supported on a BVI:
  - **arp purge-delay**
  - arp timeout
  - **bandwidth** (The default is 10 Gbps and is used as the cost metric for routing protocols for the BVI)
  - **ipv4**

- **ipv6**
- **mac-address**
- **shutdown**

- The BVI supports secondary IP addressing.
- Bi-directional egress traffic accounting is not supported on the BVI interfaces.
- BVI does not support MTU configuration using **mtu** command, which is for physical interfaces. However, **ip mtu** and **ipv6 mtu** commands, which are logical interface commands, are supported.

BVI Interface and Line Protocol States

Like typical interface states on the router, a BVI has both an Interface and Line Protocol state.

- The BVI interface state is Up when the following occurs:
  - The BVI interface is created.
  - The bridge-domain configured with the **routed interface bvi** command has at least one active bridge port available, either an Attachment Circuit or a Pseudowire.

    Attachment Circuit (AC) is a physical or logical interface that connects a customer network to a service provider network. Pseudowire (PW) is a virtual connection that emulates a physical wire, enabling data transport across a packet-switched network.



Note A BVI will be moved to the Down state if all of the bridge ports (Ethernet flow points [EFPs]) associated with the bridge domain for that BVI are down. However, the BVI will remain up if at least one bridgeport is up, even if all EFPs are down.

- These characteristics determine when the the BVI line protocol state is up:
  - The bridge-domain is in Up state.
  - The BVI IP address is not in conflict with any other IP address on another active interface in the router.

Prerequisites for Configuring IRB

You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Before configuring IRB, be sure that these tasks and conditions are met:

- Know the IP addressing and other Layer 3 information to be configured on the bridge virtual interface (BVI).

- Complete MAC address planning if you decide to override the common global MAC address for all BVIs.
- Be sure that the BVI network address is being advertised by running static or dynamic routing on the BVI interface.

Restrictions for Configuring IRB

Before configuring IRB, consider these restrictions:

- Only one BVI can be configured in any bridge domain.
- The same BVI can not be configured in multiple bridge domains.
- MTU configuration and fragmentation of packets is not supported on BVI interfaces.
- IGP ECMP path list with mix of BVI and non-BVI paths is not supported.
- The following areas are *not* supported on the Layer2 bridging (with BVI):
  - Static mac entry configuration in Bridge.
  - Mac ageing configuration at global config mode.
  - MAC Learning Disable.
  - Vlan rewrite.
- QOS configuration on BVI interface is not supported for egress.
- Label allocation mode per-CE with BVI is not supported in an access network along with PE-CE protocols enabled.

How to Configure IRB

This section includes the following configuration tasks:

Configuring the Bridge Group Virtual Interface

To configure a BVI, complete the following steps.

Configuration Guidelines

Consider the following guidelines when configuring the BVI:

- The BVI must be assigned an IPv4 or IPv6 address that is in the same subnet as the hosts in the bridged segments.
- If the bridged network has multiple IP networks, then the BVI must be assigned secondary IP addresses for each network.

Procedure

Step 1 **configure**

Example:

```
Router# configure
```

Enters the global configuration mode.

Step 2 **interface bvi** *identifier*

Example:

```
Router(config)# interface bvi 1
```

Specifies or creates a BVI, where *identifier* is a number from 1 to 65535.

Step 3 **ipv4 address** *ipv4-address mask* [**secondary**] **ipv6 address** *ipv6-prefix/prefix-length* [**eui-64**] [**route-tag** *route-tag value*]

Example:

```
Router(config-if)# ipv4 address 10.10.0.4 255.255.255.0
```

Specifies a primary or secondary IPv4 address or an IPv6 address for an interface.

Step 4 **arp purge-delay** *seconds*

Example:

```
Router(config-if)#arp purge-delay 120
```

(Optional) Specifies the amount of time (in *seconds*) to delay purging of Address Resolution Protocol (ARP) table entries when the interface goes down.

The range is 1 to 65535. By default purge delay is not configured.

Step 5 **arp timeout** *seconds*

Example:

```
Router(config-if)# arp timeout 12200
```

(Optional) Specifies how long dynamic entries learned on the interface remain in the ARP cache.

The range is 30 to 2144448000 seconds. The default is 14,400 seconds (4 hours).

Step 6 **bandwidth** *rate*

Example:

```
Router(config-if)# bandwidth 1000000
```

(Optional) Specifies the amount of bandwidth (in kilobits per second) to be allocated on the interface. This number is used as the cost metric in routing protocols for the BVI.

The range is 0 to 4294967295. The default is 10000000 (10 Gbps).

Step 7 **end** or **commit**

Example:

```
Router(config-if)# end
```

or

```
Router(config-if)# commit
```

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

  ```
  Uncommitted changes found, commit them before exiting(yes/no/cancel)?
  [cancel]:
  ```

  Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.

  Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.

  Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.

- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring the Layer 2 AC Interfaces

To configure the Layer 2 Attachment Circuit (AC) interfaces for routing by a BVI, complete these steps.

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters global configuration mode.

Step 2 **interface** [**HundredGigE** | **TenGigE**] **l2transport**

Example:

```
RP/0/RP0/CPU0:router(config)# interface TenGigE 0/0/0/0.1 l2transport
```

Enables Layer 2 transport mode on a Gigabit Ethernet or 10-Gigabit Ethernet interface or subinterface and enters interface or subinterface configuration mode.

Step 3 **end** or **commit**

Example:

```
RP/0/RP0/CPU0:router(config-if)# end
```

or

```
RP/0/RP0/CPU0:router(config-if)# commit
```

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

```
Uncommitted changes found, commit them before exiting(yes/no/cancel)?
[cancel]:
```

- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
- Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
- Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring a Bridge Group and Assigning Interfaces to a Bridge Domain

To configure a bridge group and assign interfaces to a bridge domain, complete the following steps.

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters global configuration mode.

Step 2 **l2vpn**

Example:

```
RP/0/RP0/CPU0:router(config)# l2vpn
```

Enters L2VPN configuration mode.

Step 3 **bridge group** *bridge-group-name*

Example:

```
RP/0/RP0/CPU0:router(config-l2vpn)# bridge group 10
```

Creates a bridge group and enters L2VPN bridge group configuration mode.

Step 4 **bridge-domain** *bridge-domain-name*

Example:

```
RP/0/RP0/CPU0:router(config-l2vpn-bg)# bridge-domain BD_1
```

Creates a bridge domain and enters L2VPN bridge group bridge domain configuration mode.

Step 5 **interface** [**HundredGigE | TenGigE**

Example:

```
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd)# interface HundredGigE 0/0/1/0.1
```

Associates the 100-Gigabit Ethernet or 10-Gigabit Ethernet interface with the specified bridge domain and enters L2VPN bridge group bridge domain attachment circuit configuration mode.

Repeat this step for as many interfaces as you want to associate with the bridge domain.

Step 6 **end** or **commit**

Example:

```
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd-ac)# end
```

or

```
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd-ac)# commit
```

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

```
Uncommitted changes found, commit them before exiting(yes/no/cancel)?
[cancel]:
```

- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
- Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
- Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Associating the BVI as the Routed Interface on a Bridge Domain

To associate the BVI as the routed interface on a bridge domain, complete the following steps.

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters global configuration mode.

Step 2 **l2vpn**

Example:

```
RP/0/RP0/CPU0:router(config)# l2vpn
```

Enters L2VPN configuration mode.

Step 3 **bridge group** *bridge-group-name*

Example:

```
RP/0/RP0/CPU0:router(config-l2vpn)# bridge group BG_test
```

Creates a bridge group and enters L2VPN bridge group configuration mode.

Step 4 **bridge-domain** *bridge-domain-name*

Example:

```
RP/0/RP0/CPU0:router(config-l2vpn-bg)# bridge-domain 1
```

Creates a bridge domain and enters L2VPN bridge group bridge domain configuration mode.

Step 5 **routed interface bvi** *identifier*

Example:

```
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd)# routed interface bvi 1
```

Associates the specified BVI as the routed interface for the interfaces assigned to the bridge domain.

Step 6 **end** or **commit**

Example:

```
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd)# end
```

or

```
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd)# commit
```

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

```
Uncommitted changes found, commit them before exiting(yes/no/cancel)?
[cancel]:
```

- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
- Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
- Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Displaying Information About a BVI

To display information about BVI status and packet counters, use the following commands:

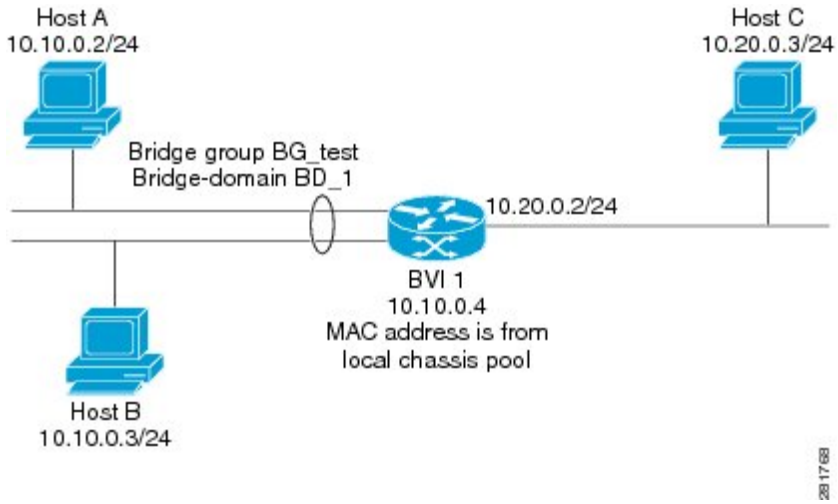
show interfaces bvi *identifier* [**accounting** | **brief** | **description** | **detail**]	Displays interface status, line protocol state, and packet counters for the specified BVI.
show adjacency bvi *identifier* [**detail** | **remote**]	Displays packet and byte transmit counters per adjacency to the specified BVI.
show l2vpn bridge-domain detail	Displays the reason that a BVI is down.

Additional Information on IRB

Packet Flows Using IRB

This figure shows a simplified functional diagram of an IRB implementation to describe different packet flows between Host A, B, and C. In this example, Host C is on a network with a connection to the same router. In reality, another router could be between Host C and the router shown.

Figure 1: IRB Packet Flows Between Hosts



When IRB is configured on a router, the following processing happens:

- ARP requests are resolved between the hosts and BVI that are part of the bridge domain.
- All packets from a host on a bridged interface go to the BVI if the destination MAC address matches the BVI MAC address. Otherwise, the packets are bridged.
- For packets destined for a host on a routed network, the BVI forwards the packets to the routing engine before sending them out a routed interface.
- All packets either from or destined to a host on a bridged interface go to the BVI first (unless the packet is destined for a host on the bridge domain).
- For packets that are destined for a host on a segment in the bridge domain that come in to the router on a routed interface, the BVI forwards the packet to the bridging engine, which forwards it through the appropriate bridged interface.

Packet Flows When Host A Sends to Host B on the Bridge Domain

When Host A sends data to Host B in the bridge domain on the 10.10.0.0 network, no routing occurs. The hosts are on the same subnet and the packets are bridged between their segment interfaces on the router.

Packet Flows When Host A Sends to Host C From the Bridge Domain to a Routed Interface

Using host information from this figure, the following occurs when Host A sends data to Host C from the IRB bridging domain to the routing domain:

- Host A sends the packet to the BVI (as long any ARP request the is resolved between the host and the BVI). The packet has the following information:
  - Source MAC address of host A.
  - Destination MAC address of the BVI.

- Since Host C is on another network and needs to be routed, the BVI forwards the packet to the routed interface with the following information:
  - IP source MAC address of Host A (10.10.0.2) is changed to the MAC address of the BVI (10.10.0.4).
  - IP destination address is the IP address of Host C (10.20.0.3).
- Interface 10.20.0.2 sees receipt of a packet from the routed BVI 10.10.0.4. The packet is then routed through interface 10.20.0.2 to Host C.

Packet Flows When Host C Sends to Host B From a Routed Interface to the Bridge Domain

Using host information from this figure, the following occurs when Host C sends data to Host B from the IRB routing domain to the bridging domain:

- The packet comes into the routing domain with the following information:
  - MAC source address—MAC of Host C.
  - MAC destination address—MAC of the 10.20.0.2 ingress interface.
  - IP source address—IP address of Host C (10.20.0.3).
  - IP destination address—IP address of Host B (10.10.0.3).
- When interface 10.20.0.2 receives the packet, it looks in the routing table and determines that the packet needs to be forwarded to the BVI at 10.10.0.4.
- The routing engine captures the packet that is destined for the BVI and forwards it to the BVI's corresponding bridge domain. The packet is then bridged through the appropriate interface if the destination MAC address for Host B appears in the bridging table, or is flooded on all interfaces in the bridge group if the address is not in the bridging table.

Configuration Examples for IRB

This section provides the following configuration examples:

Basic IRB Configuration: Example

The following example shows how to perform the most basic IRB configuration:

```
! Configure the BVI and its IPv4 address
!
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)#interface bvi 1
RP/0/RP0/CPU0:router(config-if)#ipv4 address 10.10.0.4 255.255.255.0
RP/0/RP0/CPU0:router(config-if))# exit
!
! Configure the Layer 2 AC interface
!
```

```
RP/0/RP0/CPU0:router(config)#interface HundredGigE 0/0/1/0 l2transport
RP/0/RP0/CPU0:router(config-if))# exit
!
! Configure the L2VPN bridge group and bridge domain and assign interfaces
!
RP/0/RP0/CPU0:router(config)#l2vpn
RP/0/RP0/CPU0:router(config-l2vpn)#bridge group 10
RP/0/RP0/CPU0:router(config-l2vpn-bg)#bridge-domain 1
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd)#interface HundredGigE 0/0/1/0
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd-if)# exit
!
! Associate a BVI to the bridge domain
!
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd)# routed interface bvi 1
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd)# commit
```

IRB With BVI and VRRP Configuration: Example

This example shows a partial router configuration for the relevant configuration areas for IRB support of a BVI and VRRP:

Note VRRPv6 is also supported.

```
l2vpn
 bridge group IRB
  bridge-domain IRB-EDGE
   interface TenGigE0/0/0/8
!
   routed interface BVI 100
!
interface TenGigE0/0/0/8
  l2transport
!
interface BVI 100
 ipv4 address 10.21.1.1 255.255.255.0
!
router vrrp
 interface BVI 100
 address-family ipv4
 vrrp 1
 address 10.21.1.100
 priority 100
 !
```