Bridging

This chapter contains these topics:

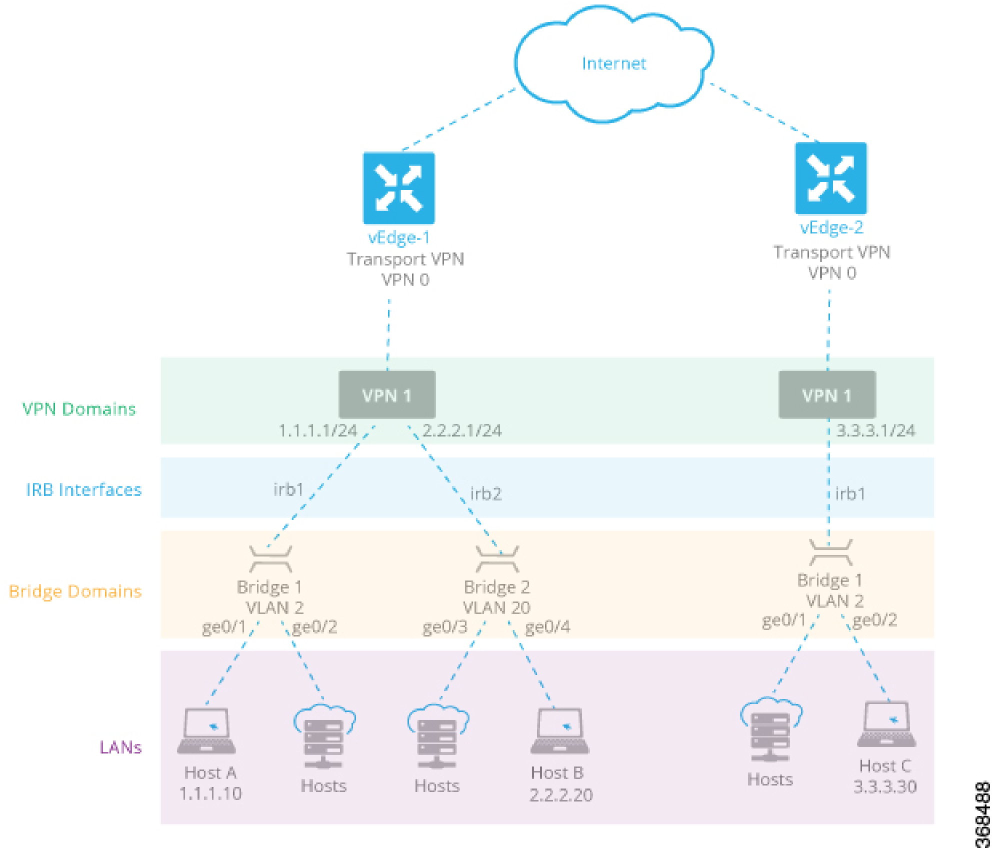
Bridging Overview

A Cisco vEdge device can act as a transparent bridge, switching traffic between LANs that are part of a VLAN at the local device's site. To implement bridging, the Cisco SD-WAN architecture defines the concept of a *bridge domain*. Each bridge domain corresponds to a single VLAN. From a switching point of view, each bridge domain is a separate broadcast domain, and each has its own Ethernet switching table (or MAC table) to use for switching traffic within the broadcast domain. Multiple bridge domains, and hence multiple VLANs, can coexist on a single Cisco vEdge device.

To allow hosts in different bridge domains to communicate with each other, Cisco vEdge devices support *integrated routing and bridging* (IRB). IRB is implemented using *logical IRB interfaces*, which connect a bridge domain to a VPN, or what might better be called a *VPN domain*. The VPN domain provides the Layer 3 routing services necessary so that traffic can be exchanged between different VLANs. Each bridge domain can have a single IRB interface and can connect to a single VPN domain, and a single VPN domain can connect to multiple bridge domains on a vEdge router. The route table in the VPN domain provides reachability between all bridge domains which participate in that VPN domain, whether the bridge domain is located on the local router or on a remote router.

Components of Bridging

The following figure illustrates the components of the Cisco SD-WAN bridging solution.



Bridge Domains

In standard transparent bridging, virtual LANs, or VLANs, segregate LANs into logical LANs, and each VLAN is an isolated broadcast domain. All VLAN traffic remains in the VLAN, and it is directed to its destination by means of Ethernet switching tables. The Cisco SD-WAN implementation of bridging overlays the concept of a *bridge domain* on top of the standard VLAN: A bridge domain comprises a single VLAN, and all the ports within a VLAN are part of a single broadcast domain. Within each broadcast domain, the standard bridging operations of learning, forwarding, flooding, filtering, and aging are performed on VLAN traffic to create and maintain the Ethernet switching table (or MAC table) for that VLAN, and hence for that bridge domain.

Each bridge domain is identified by a number. The VLAN within a bridge domain is identified by an 802.1Q identifier, which is called a VLAN tag or VLAN ID. Frames within a bridge domain can remain untagged, or you can configure a VLAN ID to tag the frames. In the Cisco SD-WAN design, the VLAN and the VLAN ID are the property of the bridge domain. They are not the property of an interface or a switching port.

Ports that connect to the WAN segments are associated with a bridge domain. In the Cisco SD-WAN overlay network, these ports are the physical Gigabit Ethernet interfaces on Cisco vEdge devices. Specifically, they are the base interfaces, for example, **ge-0/0**. You cannot use subinterfaces for bridge domain ports.

Each broadcast domain in the Cisco SD-WAN overlay network is uniquely identified by the combination of bridge domain number and VLAN ID (if configured). This design means that The same VLAN ID can be used in different bridge domains on a single Cisco vEdge device. For example, the VLAN ID 2 can exist in bridge domain 1 and bridge domain 50. In a situation where the VLAN IDs are different, two bridge domains can include the same port interfaces. For example, both (bridge 2, VLAN 2) and (bridge 10, VLAN 23) can include interfaces ge0/0 and ge0/1. Here, these two interfaces effectively become trunk ports. However, because of how interface names are tracked internally, two bridge domains that use the same VLAN ID can have no overlap between the interfaces in the two domains. For example, if (bridge 1, VLAN 2) includes interfaces ge0/0 and ge0/1, these interfaces cannot be in (bridge 50, VLAN 2).

As mentioned above, all member interfaces within a VLAN are part of a single broadcast domain. Within each broadcast domain, the standard transparent bridging operations of learning, forwarding, flooding, filtering, and aging are performed on VLAN traffic to create and maintain the Ethernet switching table, also called the MAC table, for that VLAN.

The Cisco SD-WAN bridging domain architecture lacks the concepts of access ports and trunk ports. However, the Cisco SD-WAN architecure emulates these functions. For a Cisco vEdge device that has a single bridge domain, the interfaces in the bridge emulate access ports and so the router is similar to a single switch device. For a Cisco vEdge device with multiple bridge domains that are tagged with VLAN IDs, the interfaces in the bridges emulate trunk ports, and you can think of each domain as corresponding to a separate switching device.

Native VLAN

Cisco SD-WAN bridge domains support 802.1Q native VLAN. All traffic sent and received on an interface configured for native VLAN do not have a VLAN tag in its Ethernet frame. That is, they are not tagged with a VLAN ID. If a host is connected on an interface enabled for native VLAN, the bridge domain receives no tagged frames. If the bridge domain connects to a switch that support trunk ports or connects to a hub, the bridge domain might receive both untagged and tagged frames.

Native VLAN is used primarily on trunk ports. VLAN provides backwards compatibility for devices that do not support VLAN tagging. For example, native VLAN allows trunk ports to accept all traffic regardless of what devices are connected to the port. Without native VLAN, the trunk ports would accept traffic only from devices that support VLAN tagging.

Integrated Routing and Bridging (IRB)

Bridge domains and VLANs provide a means to divide a LAN into smaller broadcast domains. Each VLAN is a separate broadcast domain, and switching within that domain directs traffic to destinations within the VLAN. The result is that hosts within a single bridge domain can communicate among themselves, but cannot communicate with hosts in other VLANs. So, for example, if a business places its departments in separate VLANs, people within the finance department would be able to communicate only with others in that department, but would not be able to communicate with the manufacturing or engineering department.

The only way for traffic to cross Layer 2 VLAN boundaries to allow communicatation between bridge domains is via Layer 3 routing. This process of marrying switching and routing is done by *integrated routing and bridging*, or IRB. With IRB, a single Cisco vEdge device can pass traffic among different bridge domains on the same router and among bridge domains on remote vEdge routers. The only restriction is that all the bridge domains must reside in the same VPN domain in the overlay network.

The Cisco SD-WAN implementation of IRB connects a Layer 2 bridge domain to a Layer 3 VPN domain via an IRB interface. An IRB interface is a logical interface that inherits all the properties of a regular interface, but it is not associated with a port or with a physical interface. Each IRB interface is named with the stem “irb” and a number that matches the number of a bridge domain. For example, the interface **irb2** is the logical interface that connects to bridge domain 2. IRB interfaces cannot have subinterfaces.

You create IRB interfaces within a VPN. A VPN domain supports multiple IRB interfaces.

There is a one-to-one association between an IRB logical interface and a bridge domain: an IRB interface can be associated only with one bridge domain, and a bridge domain can be associated with only one IRB interface. As a result, a bridge domain can be part of only one VPN in the overlay network.

The IP address of an IRB interface is the subnet of the VLAN that resides in the bridge domain. From a switching perspective, the IP address of the IRB interface is part of the bridge domain.

Configure Bridging Using Cisco vManage

To have a Cisco vEdge device act as a transparent bridge, configure bridging domains on the router. A router can have up to 16 bridging domains.

Configure Bridging and Bridge Domains

1. In Cisco vManage, select **Configuration** > **Templates**.
2. Click the **Feature** tab to view your existing feature templates or create a new one.
3. Click **Add Template**. Select a device from the list of devices. The templates available for the selected device display in the right pane.
4. Choose the **Bridge** template.
5. Enter a name and description for the template.
6. Configure bridging domains under the **Basic Configuration** tab.

Parameter Name	Description
Bridge Name	Enter a text description of the bridging domain. It can be up to 32 characters.
VLAN ID	Enter the VLAN identifier to associate with the bridging domain. *Range:* 0 through 4095
Maximum MAC Addresses	Specify the maximum number of MAC addresses that the bridging domain can learn. *Range:* 0 through 4096 *Default:* 1024
Age-Out Time (Seconds)	Specify how long to store an entry in the MAC table before it ages out. *Range:* 10 through 4096 seconds *Default:* 300 seconds (5 minutes)

7. Click **Save**.

Associate Interfaces with the Bridge Domain

To associate an interface with the bridge domain, click the Interface tab and click the **New Interface** button.

Parameter Name	Description
Interface Name	Enter the name of the interface to associate with the bridging domain, in the format **ge***slot*/*port*.
Description	Enter a text description of the interface.
Native VLAN Support	Click Enabled to configure the interface to carry untagged traffic. By default, native VLAN is disabled.
Shutdown	Click No to enable the interface. By default, an interface in a bridge domain is disabled.
Static MAC Address	Click **Add Static MAC Address**, and in the MAC Static Address field that appears, enter a static MAC address entry for the interface in the bridge domain. Click Add MAC Address to add another static MAC address entry for the interface. Click Save to save the MAC address or addresses.

Configure Interface Bridge

Integrated routing and bridging (IRB) allows Cisco vEdge devices in different bridge domains to communicate with each other. To enable IRB, create logical IRB interfaces to connect a bridge domain to a VPN. The VPN provides the Layer 3 routing services necessary so that traffic can be exchanged between different VLANs. Each bridge domain can have a single IRB interface and can connect to a single VPN, and a single VPN can connect to multiple bridge domains on a Cisco vEdge device.

1. In Cisco vManage, navigate to **Configuration** > **Templates**.
2. Click the **Feature** tab to view your existing feature templates or create a new one.
3. Click **Add Template**. Select a device from the list of devices. The templates available for the selected device display in the right pane.
4. Choose the **VPN Interface Bridge** template.
5. Enter a name and description for the template and enter the parameter. Enter other parameters described in the subsequent sections.

Create a Bridging Interface

To configure an interface to use for bridging servers, select the **Basic Configuration** tab and enter the following details.

Parameter Name	Description
Shutdown	Click No to enable the interface.

Parameter Name	Description
Interface Name	Enter the name of the interface, in the format **irb***number*. The IRB interface number can be from 1 through 63, and must be the same as the VPN identifier configured in the Bridge feature template for the bridging domain that the IRB is connected to.
Description	Enter a description for the interface.
IPv4 Address	Enter the IPv4 address of the router.
DHCP Helper	Enter up to eight IP addresses for DHCP servers in the network, separated by commas, to make the interface a DHCP helper. A DHCP helper interface forwards BOOTP (Broadcast) DHCP requests that it receives from the specified DHCP servers.
Block Non Source IP	Click Yes to have the interface forward traffic only if the source IP address of the traffic matches the interface's IP prefix range.
Secondary IP Address (Maximum: 4)	Click Add to configure up to four secondary IPv4 addresses for a service-side interface.

To save the template, click **Save**.

Apply Access Lists

To apply access lists to IRB interfaces, select the ACL tab and configure the following parameters:

Parameter Name	Description
Ingress ACL - IPv4	Click On, and specify the name of an IPv4 access list to packets being received on the interface.
Egress ACL - IPv4	Click On, and specify the name of an IPv4 access list to packets being transmitted on the interface.

To save the template, click **Save**.

Configure VRRP

To have an interface run the Virtual Router Redundancy Protocol (VRRP), which allows multiple devices to share a common virtual IP address for default gateway redundancy, select the VRRP tab. Then click the **New VRRP** button and configure the following parameters:

Parameter Name	Description
Group ID	Enter the virtual router ID, which is a numeric identifier of the virtual router. You can configure a maximum of 24 groups.*Range:* 1 through 255

Parameter Name	Description
Priority	Enter the priority level of the router. There router with the highest priority is elected as the primary device. If two Cisco vEdge devices have the same priority, the one with the higher IP address is elected as the primary one.*Range:* 1 through 254*Default:* 100
Timer	Specify how often the primary VRRP router sends VRRP advertisement messages. If the subordinate routers miss three consecutive VRRP advertisements, they elect a new primary router.*Range:* 1 through 3600 seconds *Default:* 1 second
Track OMP Track Prefix List	By default, VRRP uses of the state of the service (LAN) interface on which it is running to determine which Cisco vEdge device is the primary virtual router. If a Cisco vEdge device loses all its WAN control connections, the LAN interface still indicates that it is up even though the router is functionally unable to participate in VRRP. To take WAN side connectivity into account for VRRP, configure one of the following: Track OMP—Click **On** for VRRP to track the Overlay Management Protocol (OMP) session running on the WAN connection. If the primary VRRP router loses all its OMP sessions, VRRP elects a new default gateway from those that have at least one active OMP session. Track Prefix List—Track both the OMP session and a list of remote prefixes, which is defined in a prefix list configured on the local router. If the primary VRRP router loses all its OMP sessions, VRRP failover occurs as described for the Track OMP option. In addition, if reachability to one of the prefixes in the list is lost, VRRP failover occurs immediately, without waiting for the OMP hold timer to expire, thus minimizing the amount of overlay traffic is dropped while the Cisco IOS XE SD-WAN device determines the primary VRRP router.
IP Address	Enter the IP address of the virtual router. This address must be different from the configured interface IP addresses of both the local Cisco IOS XE SD-WAN device and the peer running VRRP.

Add ARP Table Entries

Parameter Name	Description
IP Address	Enter the IP address for the ARP entry in dotted decimal notation or as a fully qualified host name.
MAC Address	Enter the MAC address in colon-separated hexadecimal notation.

To save the ARP configuration, click **Add**.

To save the template, click **Save**.

Advanced Interface Properties

To configure other interface properties, select the Advanced tab and configure the following parameters:

Parameter Name	Description
MAC Address	Specify a MAC address to associate with the interface, in colon-separated hexadecimal notation.
IP MTU	Specify the maximum MTU size of packets on the interface. *Range:* 576 through 1804 *Default:* 1500 bytes
TCP MSS	Specify the maximum segment size (MSS) of TPC SYN packets passing through the vEdge router. By default, the MSS is dynamically adjusted based on the interface or tunnel MTU such that TCP SYN packets are never fragmented. *Range:* 552 to 1460 bytes *Default:* None
Clear-Dont-Fragment	Click On to clear the Don't Fragment (DF) bit in the IPv4 packet header for packets being transmitted out the interface. When the DF bit is cleared, packets larger than that interface's MTU are fragmented before being sent.
ARP Timeout	Specify how long it takes for a dynamically learned ARP entry to time out. *Range:* 0 through 2678400 seconds (744 hours) *Default:* 1200 seconds (20 minutes)
ICMP Redirect	Click Disable to disable ICMP redirect messages on the interface. By default, an interface allows ICMP redirect messages.

To save the template, click **Save**.

Configure Bridging Using CLI

Configure Bridging and Bridge Domains Using CLI

Bridge domains can be marked with a VLAN tag, or they can remain untagged.

Create a Bridge Domain That Uses VLAN Tagging

For a bridge domain that uses VLAN tagging, a tag, called a VLAN ID, is inserted into all frame headers sent by the domain This tag identifies which VLAN the frames belong to, and it is used to determine which interfaces the Cisco vEdge device should send broadcast packets to.

To configure a bridge domain that uses VLAN tagging, create a bridging domain, assign a VLAN tag to that domain, and associate an interface with the domain:

1. Create a bridging domain:

   ```
   vEdge(config)#  bridge bridge-id
   ```

   Each domain is identified by a unique integer, in the range 1 through 63. Each Cisco vEdge device can have up to 16 bridging domains.

2. Tag the bridging domain with a VLAN ID:

   ```
   vEdge(config-bridge)# vlan number
   ```

   The VLAN identifier can be a value from 1 through 4095.

3. Associate an interface with the bridging domain, and enable that interface:

   ```
   vEdge(config-bridge)# interface  ge slot/port
   vEdge(config-interface)# no shutdown
   ```

   The interface must be a physical interface. You cannot use subinterfaces.

After you have added physical interfaces to a VLAN, if you want to change the VLAN identifier, you must first delete all the interfaces from the VLAN. Then configure a new VLAN identifier, and re-add the interfaces to the VLAN.

You can also configure these optional parameters:

1. Configure a description for the VLAN interface, to help identify the interface in operational command output:

   ```
   vEdge(config-bridge)# interface ge slot
   /
   port
   vEdge(config-interface)# description "
   text description "
   ```

2. Configure a static MAC address for the VLAN interface:

   ```
   vEdge(config-interface)# static-mac-address aa
   :
   bb
   :
   cc
   ```

```
:
dd
:
ee
:ff
```

3. Configure a name for the VLAN, to help identify the VLAN in operational command output:

   ```
   vEdge(config-bridge)# name "text description"
   ```

4. By default, a bridging domain can learn up to 1024 MAC addresses. You can modify this to a value from 0 through 4096:

   ```
   vEdge(config-bridge)#  max-macs number
   ```

5. By default, MAC table entries age out after 300 seconds (5 minutes). You can modify this to a value from 10 through 4096 seconds:

   ```
   vEdge(config-bridge)#  age-time seconds
   ```

Here is an example configuration:

```
vEdge# config
vEdge(config)# bridge 2
vEdge(bridge-2)# vlan 27
vEdge(bridge-2)# interface ge0/4
vEdge(interface-ge0-4)# no shutdown
vEdge(interface-ge0-4)# description "VLAN tag = 27"
vEdge(interface-ge0/4)# commit and-quit
vEdge# show running-config bridge
bridge 2
 vlan 27
 interface ge0/4
  description "VLAN tag = 27"
  no native-vlan
  no shutdown
 !
!
vEdge#
```

After your have configured an interface in a bridge domain, you add or change a VLAN identifier for that domain only by first deleting the bridge domain from the configuration (with a **no bridge** *bridge-id* command) and then reconfiguing the domain with the desired interface name and VLAN tag identifier.

To see which interfaces bridging is running on, use the **show bridge interface** command:

```
vEdge# show bridge interface
                         ADMIN   OPER    ENCAP                  RX    RX      TX    TX

BRIDGE  INTERFACE  VLAN  STATUS  STATUS  TYPE   IFINDEX  MTU   PKTS  OCTETS  PKTS  OCTETS

----------------------------------------------------------------------------------------
2       ge0/4      27    Up      Up      vlan   41       1500  4     364     0     0
```

"Up" in the Admin Status column indicates that the interface has been configured, and "Up" in the Oper Status column indicates that bridging is running on the interface.

Create a Bridge Domain with an Untagged VLAN

All frames in an untagged VLAN are sent with no VLAN tag, or VLAN ID, in the frame header. For frames that already contain a tag, the tag is removed before it is sent.

In the minimal configuration for a tagged VLAN, you simply create a bridging domain that contains an interface:

1. Create a bridging domain. This domain is identified by a unique integer.

   ```
   vEdge(config)# bridge number
   ```

   On each vEdge router, you can configure up to 16 bridging domains.

2. Associate an interface with the bridging domain, and enable that interface:

   ```
   vEdge(config-bridge)# interface  interface-name
   vEdge(config-interface)# no shutdown
   ```

You can also configure the optional parameters described in the previous section.

Configure a Native VLAN

In the minimal configuration for a native VLAN, you create a bridging domain that contains an interface, and you mark that interface as a native VLAN interface:

1. Create a bridging domain. This domain is identified by a unique integer.

   ```
   vEdge(config)# bridge number
   ```

   On each vEdge router, you can configure up to 16 bridging domains.

2. Associate an interface with the bridging domain, and enable that interface:

   ```
   vEdge(config-bridge)# interface  interface-name
   vEdge(config-interface)# no shutdown
   ```

3. Enabled native VLAN on the interface:

   ```
   vEdge(config-interface)# native-vlan
   ```

You can also configure the optional parameters described in the section about creating a tagged VLAN.

Configure IRB

With bridging, all frame traffic remains within its VLAN. To allow frames to be passed among different VLANs, you enable integrated routing and bridging (IRB). To do this, you create a logical IRB interface in a VPN domain that connects to the bridge domain. Frames with destinations in other VLANs travel over the IRB interface to the VPN domain, and the Layer 3 route table is used to forward the frames toward their destination. The route table learns the routes to other IRB interfaces. With IRB, communication can be established between VLANs that are connected to the same VPN. The VLANs can be both on the local vEdge router and on a remote router.

In a minimal configuration to configure IRB, you create an IRB interface and assign it an IP address:

1. In the desired VPN, create an IRB interface:

   ```
   vEdge(config)# vpn number
   vEdge(config-vpn)# interface  irb number
   ```

   The VPN number can be any number from 1 through 65530, which correspond to service VPNs, except for 512 (which is the management VPN). You cannot place IRB interfaces in either the transport VPN (VPN 0) or the management VPN (VPN 512). The IRB interface type is **irb**. The IRB interface number is a number from 1 through 63, and it must be the same number as the the identifier of the bridging domain that the IRB is connected to. For example, if you configure a bridging domain with an identifier of 2 (with the command **bridge 2**), the IRB interface number must be 2, and so you must configure **interface irb2**.

2. Configure an IP address for the IRB interface. This address is the subnet for the VLAN in the connected bridge domain:

```
vEdge(config-irb)# ip address prefix/length
```

3. Enable the interface:

```
vEdge(config-irb)# no shutdown
```

In all respects, the logical IRB interfaces is just another interface. This means, for instance, that you can configure additional interfaces properties as desired. (Note, however, that you cannot configure autonegotiation on IRB interfaces.) It also means that you can ping a logical IRB interface from another device in the same VPN, and you can ping the interface regardless of whether a corresponding bridge exists for that IRB interface. That is, if you configure interface **irb4**, but there is no corresponding **bridge 4**, you are still able to ping **irb4**.

Here is an example IRB configuration:

```
vEdge# show running-config vpn 1
vpn 1
 interface ge0/4
  ip address 10.20.24.15/24
  no shutdown
 !
 interface irb1
  ip address 1.1.1.15/24
  no shutdown
  access-list IRB_ICMP in
  access-list IRB_ICMP out
 !
 interface irb50
  ip address 3.3.3.15/24
  no shutdown
 !
!
vEdge# show running-config vpn 2
vpn 2
 interface irb2
  ip address 2.2.2.15/24
  no shutdown
 !
!
```

To display information about the IRB interfaces, use the **show interface** command. The IRB interfaces are listed in the Interface column, and the Encapsulation Type columns marks these interfaces as "vlan".

```
vEdge# show interface

                                  IF      IF                                                               TCP
                                  ADMIN   OPER    ENCAP                                  SPEED          MSS                 RX      TX
VPN  INTERFACE  IP ADDRESS        STATUS  STATUS  TYPE   PORT TYPE  MTU   HWADDR             MBPS   DUPLEX  ADJUST  UPTIME      PACKETS PACKETS
---------------------------------------------------------------------------------------------------------------------------------------------
0    ge0/0      10.1.15.15/24     Up      Up      null   transport  1500  00:0c:29:cb:4f:9c  10     full    0       0:02:48:12  1467    1460
0    ge0/1      -                 Up      Up      null   service    1500  00:0c:29:cb:4f:a6  10     full    0       0:02:48:12  0       0
0    ge0/2      -                 Up      Up      null   service    1500  00:0c:29:cb:4f:b0  10     full    0       0:02:48:03  0       0
0    ge0/3      10.0.20.15/24     Up      Up      null   service    1500  00:0c:29:cb:4f:ba  10     full    0       0:02:48:12  0       0
0    ge0/5      -                 Up      Up      null   service    1500  00:0c:29:cb:4f:ce  10     full    0       0:02:48:03  0       0
0    ge0/6      -                 Up      Up      null   service    1500  00:0c:29:cb:4f:d8  10     full    0       0:02:48:03  0       0
0    ge0/7      10.0.100.15/24    Up      Up      null   service    1500  00:0c:29:cb:4f:e2  10     full    0       0:02:48:12  0       0
0    system     172.16.255.15/32  Up      Up      null   loopback   1500  00:00:00:00:00:00  10     full    0       0:02:48:12  0       0
1    ge0/4      10.20.24.15/24    Up      Up      null   service    1500  00:0c:29:cb:4f:c4  10     full    0       0:02:48:00  92      14
1    irb1       1.1.1.15/24       Up      Up      vlan   service    1500  00:0c:00:00:aa:00  10     full    0       0:02:48:00  1178    0
1    irb50      3.3.3.15/24       Up      Up      vlan   service    1500  00:0c:00:00:aa:00  10     full    0       0:02:48:00  0       0
2    irb2       2.2.2.15/24       Up      Up      vlan   service    1500  00:0c:00:00:aa:00  10     full    0       0:02:48:01  0       0
512  eth0       10.0.1.15/24      Up      Up      null   service    1500  00:50:56:00:01:05  1000   full    0       0:02:48:01  210     148
```

Configuration and Monitoring Commands

CLI commands for configuring and monitoring Layer 2 bridging and Layer 3 integrated routing and bridging (IRB) on Cisco vEdge routers.

Bridging Configuration Commands

Use the following commands to configure bridging on a vEdge router.

```
bridge bridge-id
  age-time seconds
  interface interface-name
    description "text description"
    native-vlan
    [no] shutdown
    static-mac-address mac-address
  max-macs number
  name text
  vlan number
```

Bridging Monitoring Commands

Use the following commands to monitor Layer 2 bridging on a vEdge router:

- **clear bridge mac** — Clear the MAC addresses that the vEdge router has learned.
- **clear bridge statistics** —Clear the bridging statistics.
- **show bridge interface** —List information about the interfaces on which bridging is configured.
- **show bridge mac** —List the MAC addresses that the vEdge router has learned.
- **show bridge table** —List the information in the bridge forwarding table.

IRB Configuration Commands

Use the following commands to configure IRB within a VPN on a vEdge router:

```
vpn vpn-id
  interface irbnumber
    access-list acl-list
    arp
      ip address ip-address mac mac-address
    arp-timeout seconds
    autonegotiate
    clear-dont-fragment
    description "text description"
    dhcp-server (on vEdge routers only)
      address-pool prefix/length
      exclude ip-address
      lease-time minutes
      max-leases number
      offer-time minutes
      options
        default-gateway ip-address
        dns-servers ip-address
        domain-name domain-name
        interface-mtu mtu
        tftp-servers ip-address
  static-lease mac-address
ip address address/subnet
mac-address mac-address
mtu bytes
[no] shutdown
tcp-mss-adjust bytes
```

IRB Monitoring Commands

Use the following commands to monitor IRB:

- **show interface** —List information about the interfaces on which IRB is enabled.