Contents

Introduction

This document helps in understanding the concept of BDI (Bridge Domain Interface) and BVI (Bridge group Virtual Interface).

BVI and BDI interfaces are routed interfaces that represent a set of interfaces that are bridged.

For example, say that you want to bridge two interfaces on the router and want them to be in the same Layer-2 broadcast domain. In this scenario, BVI/BDI interface would act as the routed interface for those two bridged physical interfaces. All the packets coming in or going out of these bridged interfaces will have to pass through the BVI/BDI interface.
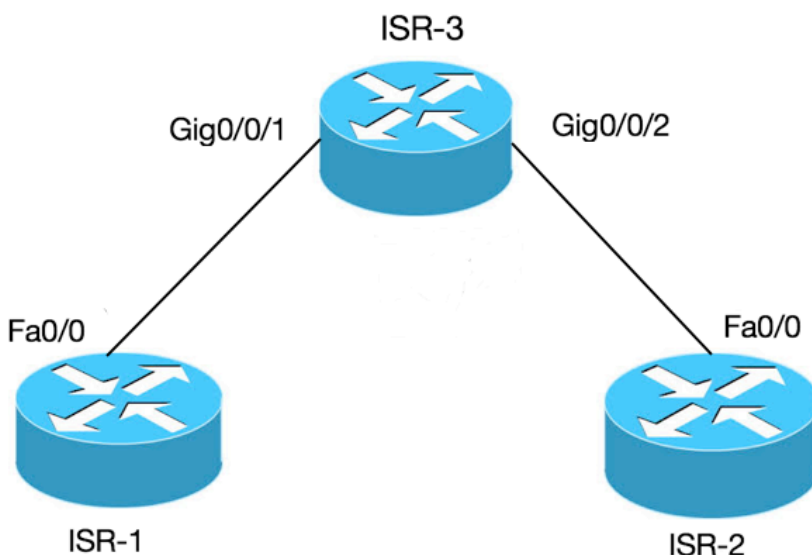
Prerequisites

Requirements

Concept of Virtual Lans.

Components Used

The information in this document is based on ISR router (for BVI) and ASR1K (for BDI).

Bridge-Group Virtual Interface: (BVI): For platforms running IOS

A router will not allow us to configure two or more Layer-3 interfaces in the same broadcast domain(Two or more interfaces in the same subnet). Let us consider a scenario where you want to connect two PCs to the router and have them part of the same subnet in addition to internet access from both the PCs.

This can be achieved using BVI concept.

Bridge-group --- Groups the physical interfaces into one logical group

Interface BVI --- Layer-3 routable logical interface

```
On ISR-3:
=========
bridge irb
bridge 1 protocol ieee
bridge 1 route ip
!
interface GigabitEthernet0/0/1
bridge-group 1
!
interface GigabitEthernet0/0/2
bridge-group 1
!
interface BVI 1
ip address 10.10.10.10
255.255.255.0
```

```
ISR-1:
=====
int fa0/0
ip address 10.10.10.1
255.255.255.0
```

```
ISR-2:
=====
int fa0/0
ip address 10.10.10.2
255.255.255.255
```

Bridge Domain Interface (BDI): For platforms running IOS-XE

This concept is very similar to BVI but for devices running IOS-XE.

Below are some of the common terminologies used :

Bridge domain represents a Layer 2 broadcast domain.

Bridge domain interface is a logical interface that allows bidirectional flow of traffic between a Layer-2 bridged network and a Layer-3 routed network.

Ethernet Virtual Circuit (EVC) is an end-to-end representation of a single instance of a Layer-2 service being offered by a provider to a customer. In the Cisco EVC Framework, the bridge domains are made up of one or more Layer-2 interfaces known as Service Instances. A Service Instance is the instantiation of an EVC on a given port on a given router. Service instance is associated with a bridge domain based on the configuration.

Here are couple of scenarios the describes the use of the Bridge Domain concept on IOS-

XE platforms:

A) Fa0/1 on both the switches are Layer-3 interfaces and are in the same broadcast domain. BDI interface configuration on ASR is **not required** if the motive is to just establish connectivity between the two switches.

```
 ASR 1K:
 ======
 interface GigabitEthernet1/2/3
 no ip address
 negotiation auto
 cdp enable
 service instance 100 ethernet
 encapsulation untagged
 bridge-domain 100
 !
 interface GigabitEthernet1/2/4
 no ip address
 negotiation auto
 cdp enable
 service instance 100 ethernet
 encapsulation untagged
 bridge-domain 100
```

```
SW1:
=====
interface FastEthernet0/1
no switchport
ip address 10.1.1.1 255.255.255.0
```

```
SW2:
=====
interface FastEthernet0/1
no switchport
ip address 10.1.1.3 255.255.255.0
```

Ping SW2 from SW1:

BGL.Q.16-3500-1#ping 10.1.1.3

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.3, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/9 ms

Note: If you want to route outside of the ASR, BDI interface configuration is required.

interface BDI100

ip address 10.1.1.2 255.255.255.0

B) Bridge multiple vlans between the two switches:

Configure separate Ethernet virtual circuits (EVC) under the physical interface for each of the vlans. Bridge-domain is not supported under the sub-interface.

Here we have two vlans. Vlan 100 and vlan 200 to be bridged:

```
 ASR 1K:                          SW1:                          SW2:
 =======                          ====                          ====
```

```
interface GigabitEthernet1/2/3
no ip address
negotiation auto
cdp enable
service instance 100 ethernet
encapsulation dot1q 100
rewrite ingress tag pop 1 symmetric
bridge-domain 100
!
service instance 200 ethernet
encapsulation dot1q 200                  interface FastEthernet0/1           interface FastEthernet0/1
rewrite ingress tag pop 1 symmetric   switchport trunk encapsulation dot1q switchport trunk encapsulation
bridge-domain 200                        switchport mode trunk              switchport mode trunk
Exact same config under Gig1/2/4      interface Vlan100                  interface Vlan100
interface GigabitEthernet1/2/4        ip address 10.1.1.1 255.255.255.0  ip address 10.1.1.3 255.255.25
no ip address                            interface Vlan200                  interface Vlan200
negotiation auto                         ip address 20.1.1.1 255.255.255.0  ip address 20.1.1.3 255.255.25
cdp enable
service instance 100 ethernet
encapsulation dot1q 100
rewrite ingress tag pop 1 symmetric
bridge-domain 100
!
service instance 200 ethernet
encapsulation dot1q 200
rewrite ingress tag pop 1 symmetric
bridge-domain 200
```

Ping int vlan100 and vlan200 on SW2 from SW1:

BGL.Q.16-3500-1#ping 10.1.1.3

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.3, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/9 ms

BGL.Q.16-3500-1#ping 20.1.1.3

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 20.1.1.3, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/9 ms