**Virtual Routing and Forwarding (VRF)-Lite**

BINGNAN XU

ccnp (cisco certified network professional)

mar. 2019

NEWPORT HIGH SCHOOL, WA

## Purpose:

The purpose of this lab is to run VRF-Lite on the three ISP routers between two sites. Left and Right are two different networks. Network traffic should not be transverses across two networks. And users in each network have no knowledge of their neighbors.

## Backgroud:

Similar to different VLAN in switches, Internet Service Providers (ISPs) sometimes are aked to separate different categories of networks. For example, though the traffic of both Company A and B go through ISP routers, company A should not have any knowledge of the data in company B. Besides separating the traffic from different customers, VRF can also be used to separate different types of traffics. For example, separate video and voice traffic from data.

Cisco Virtual Routing and Forwarding (VRF) features allows multiple virtual routers to run in one physical routers. In this way, each virtual router routes its own network (for example, Virtual Router A is private to Company A, and Virtual Router B is private to Company B. Company A and B now own their own virtual routers and traffic can be separated, though they share the same physical routers.

Though two traffics go through the same routers and switches, they should be entirely separated from each other. Each virtual router should have their distinct IP routes. IP address is supposed to be globally unique, but since two networks completely don’t have knowledge of each other, even the IP address of both sides can be identical.

## Lab summary

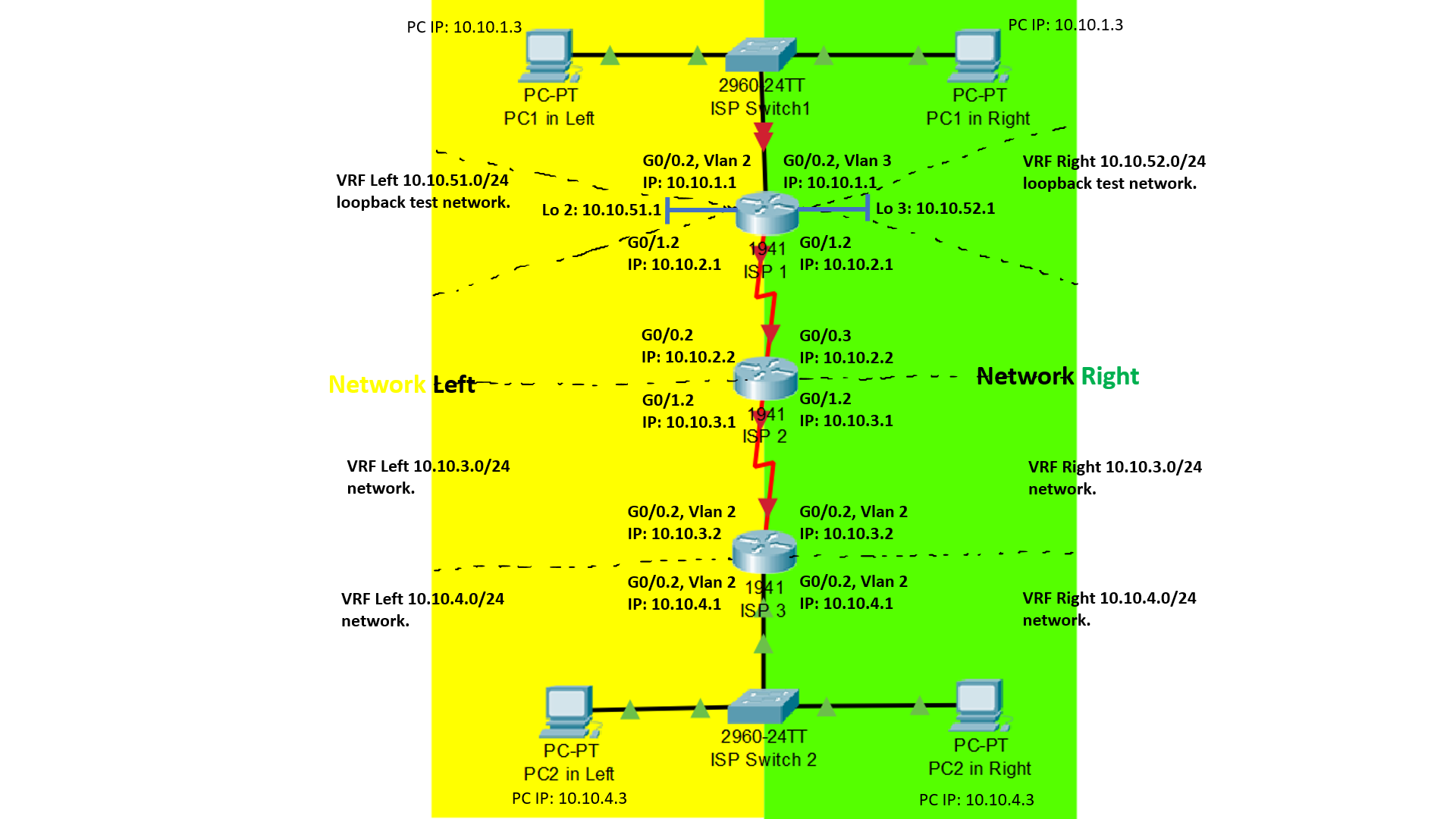
There are two theoretical companies: Company Left and Right. They are two separated companies, so their data and network traffic can’t interfere with each other.

There are three ISP routers and two shared ISP switches. The number of routers can be upscaled easily. I used VLAN 2 for all the traffic in Company Left, and VLAN 3 for Company Right (I didn’t use VLAN 1, for better security). On the routers, sub-interface G0/X.1 is for VLAN 2 and sub-interface G0/X.2 is for VLAN 3. The commands to create VRF sub-interfaces are the same as that of the Router-On-A-Stick configuration.

PC 1 in Left should ping PC 2 in Left, but not any PC in Right, vice versa. Since two networks have no knowledge of each other, I used the same IP address for the complimentary networks between Left and Right. I configure different IP addresses for Loopback 2 and Lookback 3, so that I can distinguish network Left and Right and make sure that they are not connected.

Though PC 1 in Left and PC1 in Right are connected to the same switch, they are not connected to each other. Their switchports belong to different VLANs and different VLANs forward different VRF traffics.

## Lab Diagram



## Lab commands

**Router(config)# ip vrf** *left* (words)

**Router(config-if)# ip vrf forwarding** *left* (name of the VRF)

**Router(config-if)# encapsulation dot1Q** 2 (number of the VLAN)

**Router(config)# router ospf** 2 (any OSPF process number) **vrf** left (VRF name)

**Switch(config)# ip vrf forwarding**

**Switch(config)# switchport trunk encapsulation dot1q**

## configuration of both switches and routers

### ISP Routers configuration:

**Router(config)#** **hostname R1**

ip vrf left

ip vrf right

interface Loopback2

description left

ip vrf forwarding left

ip address 10.10.51.1 255.255.255.0

interface Loopback3

description right

ip vrf forwarding right

ip address 10.10.52.1 255.255.255.0

interface GigabitEthernet0/0

no ip address

interface GigabitEthernet0/0.1

encapsulation dot1Q 2

ip vrf forwarding left

ip address 10.10.1.1 255.255.255.0

interface GigabitEthernet0/0.2

encapsulation dot1Q 3

ip vrf forwarding right

ip address 10.10.1.1 255.255.255.0

interface GigabitEthernet0/1

no ip address

interface GigabitEthernet0/1.1

encapsulation dot1Q 2

ip vrf forwarding left

ip address 10.10.2.1 255.255.255.0

interface GigabitEthernet0/1.2

encapsulation dot1Q 3

ip vrf forwarding right

ip address 10.10.2.1 255.255.255.0

router ospf 2 vrf left

network 10.10.1.0 0.0.0.255 area 0

network 10.10.2.0 0.0.0.255 area 0

network 10.10.51.0 0.0.0.255 area 0

network 10.10.52.0 0.0.0.255 area 0

router ospf 3 vrf right

network 10.10.1.0 0.0.0.255 area 0

network 10.10.2.0 0.0.0.255 area 0

network 10.10.51.0 0.0.0.255 area 0

network 10.10.52.0 0.0.0.255 area 0

end

**Note:** I put in the OSPF network statement for both lo1 and lo2. Though each sub-interface can only ping one of the lo1 or lo2, I did so to make sure that network Left and Right are indeed isolated by VRF, not because I didn’t put in the network statement.

**Router(config)#** **hostname R3**

ip vrf left

ip vrf right

interface GigabitEthernet0/0

no ip address

interface GigabitEthernet0/0.1

encapsulation dot1Q 2

ip vrf forwarding left

ip address 10.10.3.2 255.255.255.0

interface GigabitEthernet0/0.2

encapsulation dot1Q 3

ip vrf forwarding right

ip address 10.10.3.2 255.255.255.0

interface GigabitEthernet0/1

no ip address

interface GigabitEthernet0/1.1

encapsulation dot1Q 2

ip vrf forwarding left

ip address 10.10.4.1 255.255.255.0

interface GigabitEthernet0/1.2

encapsulation dot1Q 3

ip vrf forwarding right

ip address 10.10.4.1 255.255.255.0

router ospf 2 vrf left

network 10.10.3.0 0.0.0.255 area 0

network 10.10.4.0 0.0.0.255 area 0

router ospf 3 vrf right

network 10.10.3.0 0.0.0.255 area 0

network 10.10.4.0 0.0.0.255 area 0

**Router(config)#** **hostname R2**

ip vrf left

ip vrf right

interface GigabitEthernet0/0

no ip address

interface GigabitEthernet0/0.1

encapsulation dot1Q 2

ip vrf forwarding left

ip address 10.10.2.2 255.255.255.0

interface GigabitEthernet0/0.2

encapsulation dot1Q 3

ip vrf forwarding right

ip address 10.10.2.2 255.255.255.0

interface GigabitEthernet0/1

no ip address

interface GigabitEthernet0/1.1

encapsulation dot1Q 2

ip vrf forwarding left

ip address 10.10.3.1 255.255.255.0

interface GigabitEthernet0/1.2

encapsulation dot1Q 3

ip vrf forwarding right

ip address 10.10.3.1 255.255.255.0

router ospf 2 vrf left

network 10.10.2.0 0.0.0.255 area 0

network 10.10.3.0 0.0.0.255 area 0

router ospf 3 vrf right

network 10.10.2.0 0.0.0.255 area 0

network 10.10.3.0 0.0.0.255 area 0

### ISP Switches configuration:

**Switch(config)#** **hostname S2**

The rest is the same as Switch 1.

**Switch(config)#** **hostname S1**

ip vrf forwarding

ip vrf left

ip vrf right

vlan 2

**Note:** there is also an option that makes layer 3 switches routers. I used switches as layer 2 switches. If the switch needs to serve routing purposes, just use the same switch config in addition to the router configuration.

name forleft

vlan 3

name forright

interface FastEthernet1/0/1

switchport trunk encapsulation dot1q

switchport mode trunk

interface FastEthernet1/0/2

description left computer

switchport access vlan 2

switchport mode access

ip vrf forwarding left

interface FastEthernet1/0/3

description right computer

switchport access vlan 3

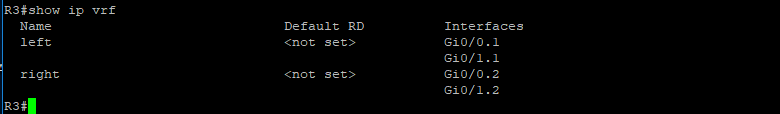
switchport mode access

ip vrf forwarding right

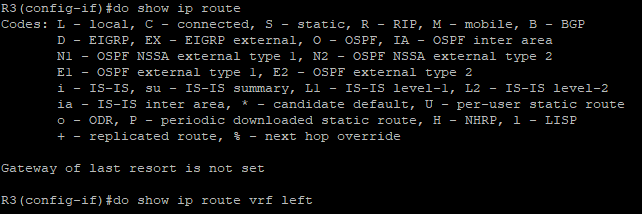
## Show commands

### verifying that routers are running VRf:

These are the 4 sub-interfaces (two on each physical port) on R3.

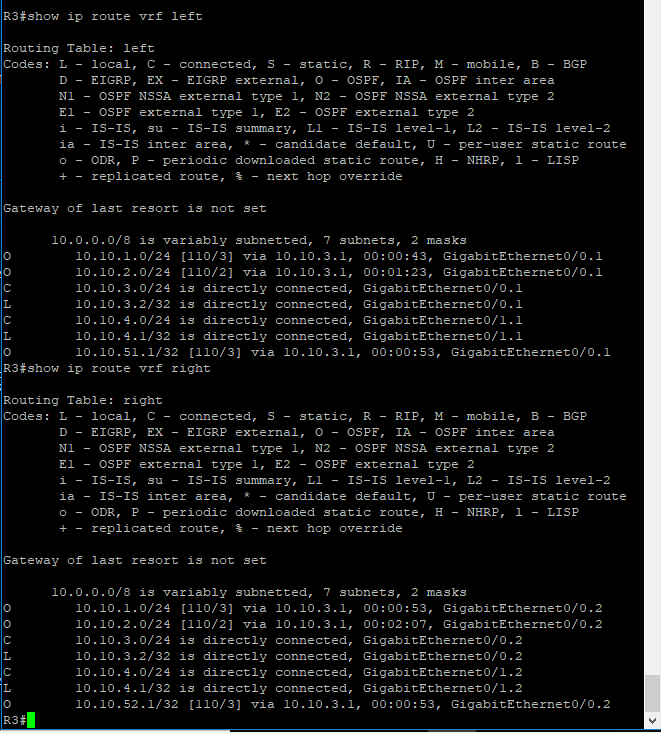


Since VRF is running on R3, the normal non-VRF router is not functional. Thus, you can’t see any normal routes on R3 (not even R3’s local ports like G0/0, G0/1, or serial ports).



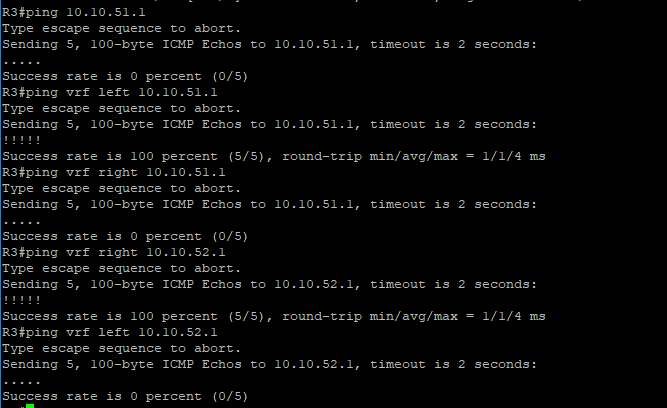
### verifying that vrf is working and two networks are isolated:

Though normal routes are not showed, VRF routes are indeed working, as showed below. VRF Left and Right are separated. It is verified by loopback R1’s 10.10.51.0/24 in Left and R2’s 10.10.52.0/24 in Right.



The next screenshot shows pings towards **loopback2 10.10.51.1 for Left** and **loopback3 10.10.52.1 for Right**. Since the router is running VRF, the normal router ping to 10.10.51.1 didn’t work. Ping from **VRF Left** to **10.10.51.1** works because loopback 2 is in VRF Left. Ping from **VRF Right** didn’t work because VRF Right and loopback 2 are in different VRF domain. It shows that my VRF configuration is indeed working and two networks are separated.

Vice versa, the same principle applies to loopback3 in VRF Right.



## Problems encountered

**Can’t issue “switchport mode trunk” command on Layer 3 Switch:**

For layer 3 switches, every port is by default a router port, which make “switchport” commands not work. Thus, to make router ports do trunk port, we need to use command

“**switchport trunk encapsulation dot1q**”.

The dot1q encapsulation makes a port switchport trunk and transmit different VLAN traffics.

**Can’t issue “router ospf area 0 vrf Left” command:**

R2 gave an error message when I issued this command. I didn’t figure out the source of the error, but I reloaded the router and this problem is fixed.

**Confusion over whether it’s a connection problem or OSPF routing problem:**

It’s difficult to troubleshot when there are multiple sources or problems. Thus, made two model: one tests only the switch Router-On-A-Stick with VRF, and another one tests the VRF connection between different routers. In the actual project, I started building my network from 10.10.1.0/24. First, make sure that PC1s on both sides can ping their loopback network, and that PC1s from both sides cannot ping each other. Then, I moved onto R2, made sure that the OSPF works, so on. I knew what works and where the problem could be. This style of working is useful in my future projects.

## Summary

VRF creates multiple virtual routers within one physical router, so that the routes and traffics within each virtual router doesn’t interfere with each other. VRF is great for separating the traffic between different users or different data categories (data, voice, video).

To test the concept that different VRF networks have no knowledge of each other, I use the same ip addresses for the complimentary sites of two companies. OSPF is my choice of routing protocol (other protocols like RIP or EIGRP would also work).

I associate a VLAN with a VRF by restricting both VLAN and VRF on one interface.

For example, on a router, these two commands associate VRF left and VLAN 2 on an interface:

**Router(config-if)# ip vrf forwarding** *left* (name of the VRF)

**Router(config-if)# encapsulation dot1Q** *2* (number of the VLAN)

On a switch, these three commands associate VRF left and VLAN 2 on an interface:

**Switch(config-if)# switchport access vlan** *2* (name of the VRF)

Switch(config-if)# switchport mode access

**Switch(config-if)# ip vrf forwarding *left*** (name of the VRF)

That’s the core of VRF commands. Then add OSPF and other necessary commands, and a VRF router is ready to go.