**SPLITTING TRAFFIC WITH VIRTUAL ROUTING AND FORWARDING WITH EIGRP**

******

***BY: ALEKSIE SANCHEZ MEDEIROS***

**

***Purpose***

To understand how VRF lite can be used to separate traffic between three point-to-point routers using EIGRP to create routes.

***Background***

Virtual Routing and Forwarding (VRF) is used to split a router’s routing table into multiple virtual routing tables. The version of VRF used in this lab is VRF lite. Originally VRF was created to be used with Multiprotocol Label and Switching (MPLS), a routing technique mostly used in large scale networks which basically labels packets on a network and uses that label to determine which path the packet should take to get to an end device. Eventually, VRF became so important in small to medium sized networks which wouldn’t need MPLS that VRF lite was created. VRF lite functions very similar to how VLANS work between switches, Created VRFs can then be applied onto an interface and used to separate traffic between two networks using the same routers. This separation of networks even allows for the same subnet addresses to be used without any issues. However, VRF also requires routes in order to be able to send packets across the network. This can be accomplished automatically with EIGRP which is a Cisco proprietary distance vector routing protocol that accomplishes exactly what we need. By using EIGRP we can ensure that there is proper network convergence without requiring multiple static routes across the whole network.

***Lab summary***

By the end of the lab, I had configured IPv4 and IPv6 connectivity across the network for the PCs and Loopbacks that were in the same VRF group and ensured that devices on other VRF groups could not intercommunicate.

***Lab Commands***

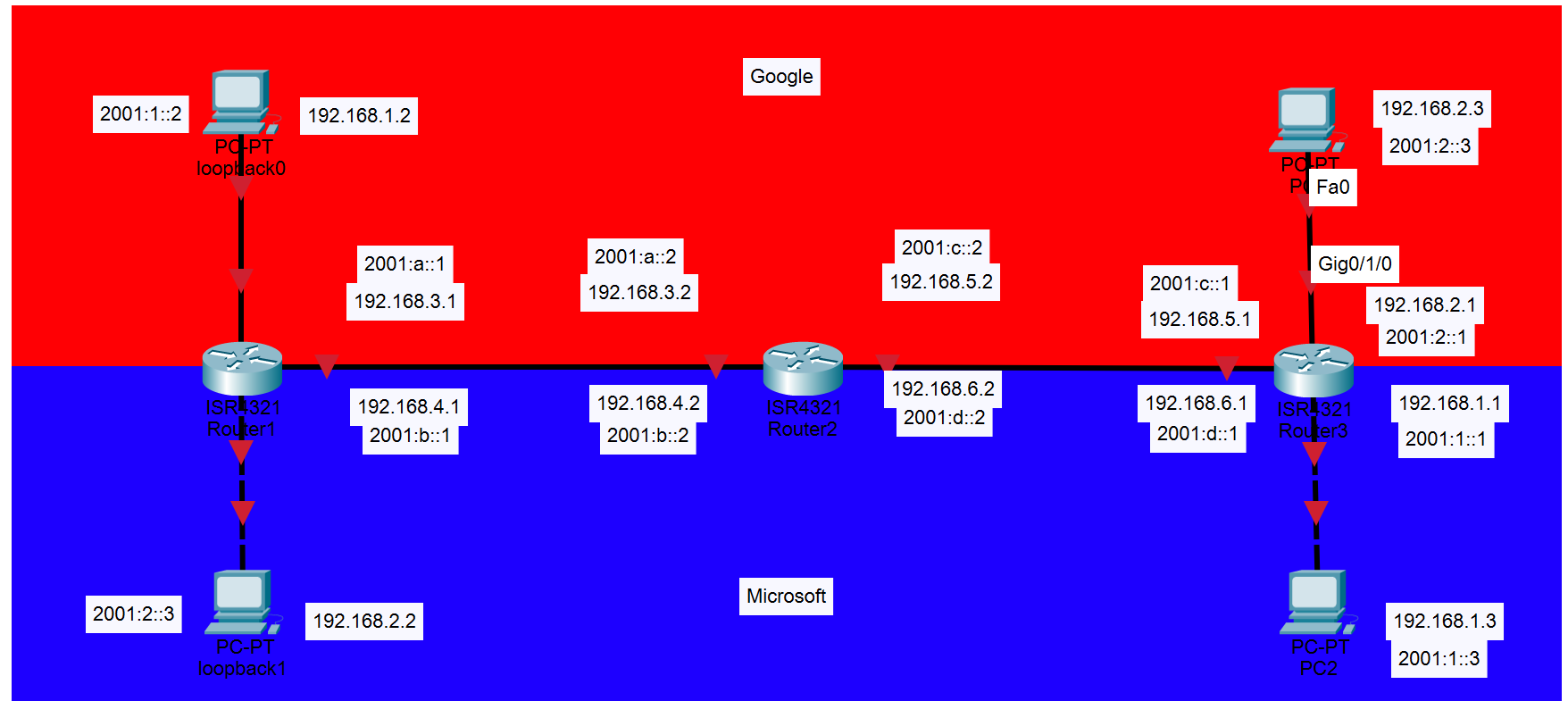
***vrf definition [name]*** is used to define VRF groups, this differs from ***ip vrf [name]*** in that it allows for IPV4 and IPV6 routing and forwarding.

***vrf forwarding [name]*** *i*s used on the port that is participating in VRF lite of the defined name, this ensures that only the information that should be shared with that network is given.

***address-family [ipv4/6] unicast vrf [name] autonomous-system [ASN]*** initializes and allows for configuration of IPv4 or IPv6 routes being shared using EIGRP and split by VRF lite.

***show ip route vrf [name]*** is used to verify that the current routers are being shared by showing the routing table associated with that VRF group.

***Network Diagram***

******  
***Router Configurations***

***R1:***

hostname R1

boot-start-marker

boot-end-marker

vrf definition Google

address-family ipv4

exit-address-family

address-family ipv6

exit-address-family

vrf definition Mgmt-intf

address-family ipv4

exit-address-family

address-family ipv6

exit-address-family

vrf definition Microsoft

address-family ipv4

exit-address-family

address-family ipv6

exit-address-family

no aaa new-model

login on-success log

subscriber templating

vtp domain cisco

vtp mode transparent

ipv6 unicast-routing

multilink bundle-name authenticated

crypto pki trustpoint TP-self-signed-859896477

enrollment selfsigned

subject-name cn=IOS-Self-Signed-Certificate-859896477

revocation-check none

rsakeypair TP-self-signed-859896477

crypto pki certificate chain TP-self-signed-859896477

certificate self-signed 01

3082032E 30820216 A0030201 02020101 300D0609 2A864886 F70D0101 05050030

30312E30 2C060355 04031325 494F532D 53656C66 2D536967 6E65642D 43657274

69666963 6174652D 38353938 39363437 37301E17 0D323231 30313731 38323635

385A170D 33303031 30313030 30303030 5A303031 2E302C06 03550403 1325494F

532D5365 6C662D53 69676E65 642D4365 72746966 69636174 652D3835 39383936

34373730 82012230 0D06092A 864886F7 0D010101 05000382 010F0030 82010A02

82010100 CA31EE51 C97FF58C 76C72B4E 2B6CD51B 98CBA177 7EEF8D11 DAAB7CA8

47B3AA97 3B815AD1 09F637AE B1D98BB8 A2CAA1A9 0AFAF87A 3AFBFF9E 34875D72

0BD5EE8D E40F4D4A 3B4A38A7 09F1940D 013C18AE F29F2BEA 07085EB5 982E9BC8

F99C8CA7 1C7DD58E 67B89FCB 951C3C4C 6D11B8C7 8D24BF5C 973A32BF E16A3094

99E8DB22 7FEA5A30 6E9457F6 90485336 E953F3D2 942824E3 87D8DE52 E00336AC

09CA85F0 0BD105FA B4078F96 9A2EA846 C147AD42 B08CD3D2 16A06EB1 CC54E167

8F4677E9 2663D37D 7B1C3891 9ABF4B5B 83ECE428 AD426108 357B992E 792C850D

84C67187 BF0E10B5 B1D23A97 F2F1372F 7D0FA8C8 80E947DE 5E0FA234 7FA6A487

24A0DB83 02030100 01A35330 51300F06 03551D13 0101FF04 05300301 01FF301F

0603551D 23041830 168014E7 C71AF39E FCC743E7 C7395603 DBBCA771 4C734E30

1D060355 1D0E0416 0414E7C7 1AF39EFC C743E7C7 395603DB BCA7714C 734E300D

06092A86 4886F70D 01010505 00038201 010029B2 769B6033 C71585B8 DD1EE596

BDB3F81C 5C58921E AF7FBE2F A95F447D 7B870BCD B9AE5E5D 46FCE0E1 667295B7

4668DACB F848F91A 207FC6CD 203E64BF 6747B9E7 6FF304F1 491442EA 56EEBEE6

DE79EC87 F5BE7B9C B2482264 A58FAC1B 827F66C7 F16C0292 815AD1ED 86F2E167

9568FC20 9E2ADCB6 311B34E4 E93EC128 2DD25078 4F27E1F1 4DD309BA B2A0248A

C41F66C8 4A81C2B8 9D0E8A62 4E0443F6 F28B3203 28A14D43 0E06A98B 06DAB16D

66E0616A DB63132A 8FB53D9B 88A28660 F84CD05D EC8653F6 C3FC6446 94977DAC

0ED87E1C 9C0B372A 6E25729B FAD2B249 6FDF7BC6 3218B110 D167D3D5 AEACB17D

6E8CB48E ED168D18 8D9104DE BA9F3515 5662

quit

license udi pid ISR4321/K9 sn FLM240608PJ

no license smart enable

diagnostic bootup level minimal

spanning-tree extend system-id

redundancy

mode none

interface Loopback0

vrf forwarding Google

ip address 192.168.1.2 255.255.255.0

ipv6 address 2001:1::2/64

interface Loopback1

vrf forwarding Microsoft

ip address 192.168.2.2 255.255.255.0

ipv6 address 2001:2::3/64

interface GigabitEthernet0/0/0

no ip address

negotiation auto

interface GigabitEthernet0/0/0.1

encapsulation dot1Q 1 native

vrf forwarding Google

ip address 192.168.3.1 255.255.255.0

ipv6 address 2001:A::1/64

interface GigabitEthernet0/0/0.2

encapsulation dot1Q 2

vrf forwarding Microsoft

ip address 192.168.4.1 255.255.255.0

ipv6 address 2001:B::1/64

interface GigabitEthernet0/0/1

no ip address

shutdown

negotiation auto

interface GigabitEthernet0/1/0

no ip address

shutdown

negotiation auto

interface GigabitEthernet0/1/1

no ip address

shutdown

negotiation auto

interface GigabitEthernet0

vrf forwarding Mgmt-intf

no ip address

shutdown

negotiation auto

router eigrp Google

address-family ipv4 unicast vrf Google autonomous-system 1

topology base

exit-af-topology

network 192.168.1.0

network 192.168.3.0

exit-address-family

address-family ipv6 unicast vrf Google autonomous-system 1

topology base

exit-af-topology

eigrp router-id 1.1.1.1

exit-address-family

router eigrp Microsoft

address-family ipv4 unicast vrf Microsoft autonomous-system 1

topology base

exit-af-topology

network 192.168.2.0

network 192.168.4.0

exit-address-family

address-family ipv6 unicast vrf Microsoft autonomous-system 1

topology base

exit-af-topology

eigrp router-id 1.1.1.1

exit-address-family

ip forward-protocol nd

ip http server

ip http authentication local

ip http secure-server

ip tftp source-interface GigabitEthernet0

control-plane

line con 0

transport input none

stopbits 1

line aux 0

stopbits 1

line vty 0 4

login

end

***R2:***

version 16.9

service timestamps debug datetime msec

service timestamps log datetime msec

platform qfp utilization monitor load 80

platform punt-keepalive disable-kernel-core

hostname R2

boot-start-marker

boot-end-marker

vrf definition Google

address-family ipv4

exit-address-family

address-family ipv6

exit-address-family

vrf definition Mgmt-intf

address-family ipv4

exit-address-family

address-family ipv6

exit-address-family

vrf definition Microsoft

address-family ipv4

exit-address-family

address-family ipv6

exit-address-family

no aaa new-model

login on-success log

subscriber templating

vtp domain cisco

vtp mode transparent

ipv6 unicast-routing

multilink bundle-name authenticated

crypto pki trustpoint TP-self-signed-4288135047

enrollment selfsigned

subject-name cn=IOS-Self-Signed-Certificate-4288135047

revocation-check none

rsakeypair TP-self-signed-4288135047

crypto pki certificate chain TP-self-signed-4288135047

certificate self-signed 01

30820330 30820218 A0030201 02020101 300D0609 2A864886 F70D0101 05050030

31312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D 43657274

69666963 6174652D 34323838 31333530 3437301E 170D3233 30313231 30303132

35385A17 0D333030 31303130 30303030 305A3031 312F302D 06035504 03132649

4F532D53 656C662D 5369676E 65642D43 65727469 66696361 74652D34 32383831

33353034 37308201 22300D06 092A8648 86F70D01 01010500 0382010F 00308201

0A028201 01009668 C5A7090F 10D0E26B E4D45357 1ABCCA03 FBCAD934 5D3F100E

E1971941 AD12727B A09983D1 A3CED08E 1F6B6109 D9D2B80B DC9BD6FF 69EB91AA

257D0A7B A864C738 03848ED1 6B6EC5F2 9FBEBE31 5932908D B291FABC 1FF89C2A

AEAD39D9 5C17951E ADB72385 016D32DD 73A38547 94BA75EF 9522F133 C05DFF7A

DAD3F3F0 82D6DC5D 3E3EB43A 46D3B39B C75978A2 015B88EF 7528435D 69D2FF1F

62695045 B1FEFFF2 A65B14B2 BDE814CE BCE0C3ED 16B6AC9E 90FC66CB B98C1F55

48DE5AC1 16166126 06DB4910 99EDA139 5B815F32 5745DC3E 89F55F0A F593D854

481CFAA5 855FFDEE 81C22640 83D9C868 7221240B 7CF466FF 265C6FD1 ADE3B39D

3373DFEC A0350203 010001A3 53305130 0F060355 1D130101 FF040530 030101FF

301F0603 551D2304 18301680 149ABBC9 19083860 4F16D5C6 55D0705F D455B426

3D301D06 03551D0E 04160414 9ABBC919 0838604F 16D5C655 D0705FD4 55B4263D

300D0609 2A864886 F70D0101 05050003 82010100 91157B2D 91377716 B5ABC3F7

6CE93551 365B5FB4 38A16153 FCC45AE8 44E2BBF3 6F741AE4 1242693F 611ACA35

F9830491 318E6093 81183845 DAA8CFC2 0108D703 4F58C75E 266F7656 5F10CBD7

CE5195D4 F50B677A 270B0728 45E485F5 FB19EE77 99E5D2E5 690F39C3 64FA3DE7

77913D3C 2C5D4F4B 0AAE30EE 3BEE72DD CB9A273A CF5957FF 38643F0F 340CED35

D45BF9AE E62ECD08 114AE73D EDBD9E69 D067D47A 315518E8 4A2722C5 ADEB3A9D

83694EB9 C0D89D8E 0333DAEE C64F6B6C 62E2B329 230570EF C5608BE0 34A5955A

B4D97C99 9BB0B8D6 C4F73C9C 41CE0F62 CD7FD197 CB6251A7 23173F72 03BDBDC3

FCAB0FB8 53454E6E D4EFF04C 562FD727 BC5FE3EA

quit

license udi pid ISR4321/K9 sn FLM2406090M

no license smart enable

diagnostic bootup level minimal

spanning-tree extend system-id

redundancy

mode none

interface GigabitEthernet0/0/0

no ip address

negotiation auto

interface GigabitEthernet0/0/0.1

encapsulation dot1Q 1 native

vrf forwarding Google

ip address 192.168.5.2 255.255.255.0

ipv6 address 2001:C::2/64

interface GigabitEthernet0/0/0.2

encapsulation dot1Q 2

vrf forwarding Microsoft

ip address 192.168.6.2 255.255.255.0

ipv6 address 2001:D::2/64

interface GigabitEthernet0/0/1

no ip address

negotiation auto

interface GigabitEthernet0/0/1.1

encapsulation dot1Q 1 native

vrf forwarding Google

ip address 192.168.3.2 255.255.255.0

ipv6 address 2001:A::2/64

interface GigabitEthernet0/0/1.2

encapsulation dot1Q 2

vrf forwarding Microsoft

ip address 192.168.4.2 255.255.255.0

ipv6 address 2001:B::2/64

interface GigabitEthernet0/1/0

no ip address

shutdown

negotiation auto

interface GigabitEthernet0/1/1

no ip address

shutdown

negotiation auto

interface GigabitEthernet0

vrf forwarding Mgmt-intf

no ip address

shutdown

negotiation auto

router eigrp Google

address-family ipv4 unicast vrf Google autonomous-system 1

topology base

exit-af-topology

network 192.168.3.0

network 192.168.5.0

eigrp router-id 2.2.2.2

exit-address-family

address-family ipv6 unicast vrf Google autonomous-system 1

topology base

exit-af-topology

eigrp router-id 2.2.2.2

exit-address-family

router eigrp Microsoft

address-family ipv6 unicast vrf Microsoft autonomous-system 1

topology base

exit-af-topology

eigrp router-id 2.2.2.2

exit-address-family

address-family ipv4 unicast vrf Microsoft autonomous-system 1

topology base

exit-af-topology

network 192.168.4.0

network 192.168.6.0

exit-address-family

ip forward-protocol nd

ip http server

ip http authentication local

ip http secure-server

ip tftp source-interface GigabitEthernet0

control-plane

line con 0

transport input none

stopbits 1

line aux 0

stopbits 1

line vty 0 4

login

end

***R3:***

hostname R3

boot-start-marker

boot-end-marker

vrf definition Google

address-family ipv4

exit-address-family

address-family ipv6

exit-address-family

vrf definition Mgmt-intf

address-family ipv4

exit-address-family

address-family ipv6

exit-address-family

vrf definition Microsoft

address-family ipv4

exit-address-family

address-family ipv6

exit-address-family

no aaa new-model

login on-success log

subscriber templating

ipv6 unicast-routing

multilink bundle-name authenticated

crypto pki trustpoint TP-self-signed-2667303412

enrollment selfsigned

subject-name cn=IOS-Self-Signed-Certificate-2667303412

revocation-check none

rsakeypair TP-self-signed-2667303412

crypto pki certificate chain TP-self-signed-2667303412

certificate self-signed 01

30820330 30820218 A0030201 02020101 300D0609 2A864886 F70D0101 05050030

31312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D 43657274

69666963 6174652D 32363637 33303334 3132301E 170D3233 30313233 31383535

33385A17 0D333030 31303130 30303030 305A3031 312F302D 06035504 03132649

4F532D53 656C662D 5369676E 65642D43 65727469 66696361 74652D32 36363733

30333431 32308201 22300D06 092A8648 86F70D01 01010500 0382010F 00308201

0A028201 0100BA48 12552FC5 762646F1 353CD39D 1B638255 9B4D3A5D 3A564568

76DB86CF 8F05447C 8ECC380C D6132E26 5E20BBFB 71DA1C57 A2B22975 3365E8C5

CC3B0647 9F7D1424 DB6CFBC1 9BD79C61 70D6EE20 9ACC193D 93AE2C69 42064583

53A0BAC2 C52C317A 04A70870 588FE9C2 0DC721C7 C3D194B3 E8053C54 C13E1ED4

A5E15F06 9FA482EC ECA458A3 E4EE6371 3CA8809B 320DEBDA EED89B16 ECC918DE

B668B7D8 1782A037 C78EC910 42610115 BF90DC1B 3C92A982 64FCBCD7 4959630C

FC1E677B 0BEDC37F 3A2384D3 1E04633C 0B54B2CB 12005C2C 15448BD2 756863E1

C163FAD4 DE06F2E6 BAE73D79 35FEB192 186347E5 6CF0E9CC E5997858 2033BAFC

BA6E00E2 BFD30203 010001A3 53305130 0F060355 1D130101 FF040530 030101FF

301F0603 551D2304 18301680 146A34F9 70113FBA 9EA49D32 27AD3186 505879C3

64301D06 03551D0E 04160414 6A34F970 113FBA9E A49D3227 AD318650 5879C364

300D0609 2A864886 F70D0101 05050003 82010100 5ED2DBAE 8FB7F8B1 662DD28E

88AE7FDC B8D14C2C 6695B6DD FD7B785E ACFA911A 7906F8FA 067192D2 00568BDF

17403348 200F7090 E943789D 7F4B4889 075B09F4 F047F53D 948F0AE9 66D0F0F0

9443379D 5C82915B CD1FB312 C2768669 DA80FE57 38413C0C BA39DF49 C2F3C096

E1772829 83E1DEE9 E824AC1C E8FEAF41 0E0F8D0B D4D72EBE 0DA4A9A0 7B3978CD

71B219BD EF2DD29C CBAC1BA1 12D7093F 7140EC78 B5180A38 01075AA2 C0E6E060

29B5390E 04AF1382 6EE75332 88FD7FF3 5ADA9F9C 7FA34369 8D341150 6AD2E682

42AE18D5 E075E68D 149F9961 1B97809B 41659B4F 1D100CA3 534DFA72 65CBFA4E

7DB927C1 4CDA2DCF 04AC3FA3 CCFFAE5A 7DABF926

quit

license udi pid ISR4321/K9 sn FLM2407011F

no license smart enable

diagnostic bootup level minimal

spanning-tree extend system-id

redundancy

mode none

interface GigabitEthernet0/0/0

no ip address

negotiation auto

interface GigabitEthernet0/0/1

no ip address

negotiation auto

interface GigabitEthernet0/0/1.1

encapsulation dot1Q 1 native

vrf forwarding Google

ip address 192.168.5.1 255.255.255.0

ipv6 address 2001:C::1/64

interface GigabitEthernet0/0/1.2

encapsulation dot1Q 2

vrf forwarding Microsoft

ip address 192.168.6.1 255.255.255.0

ipv6 address 2001:D::1/64

interface GigabitEthernet0/1/0

vrf forwarding Google

ip address 192.168.2.1 255.255.255.0

negotiation auto

ipv6 address 2001:2::1/64

interface GigabitEthernet0/1/1

vrf forwarding Microsoft

ip address 192.168.1.1 255.255.255.0

negotiation auto

ipv6 address 2001:1::1/64

interface GigabitEthernet0

vrf forwarding Mgmt-intf

no ip address

shutdown

negotiation auto

router eigrp Google

address-family ipv4 unicast vrf Google autonomous-system 1

topology base

exit-af-topology

network 192.168.2.0

network 192.168.5.0

exit-address-family

address-family ipv6 unicast vrf Google autonomous-system 1

topology base

exit-af-topology

eigrp router-id 3.3.3.3

exit-address-family

router eigrp Microsoft

address-family ipv6 unicast vrf Microsoft autonomous-system 1

topology base

exit-af-topology

eigrp router-id 3.3.3.3

exit-address-family

address-family ipv4 unicast vrf Microsoft autonomous-system 1

topology base

exit-af-topology

network 192.168.1.0

network 192.168.6.0

exit-address-family

ip forward-protocol nd

ip http server

ip http authentication local

ip http secure-server

ip tftp source-interface GigabitEthernet0

control-plane

line con 0

transport input none

stopbits 1

line aux 0

stopbits 1

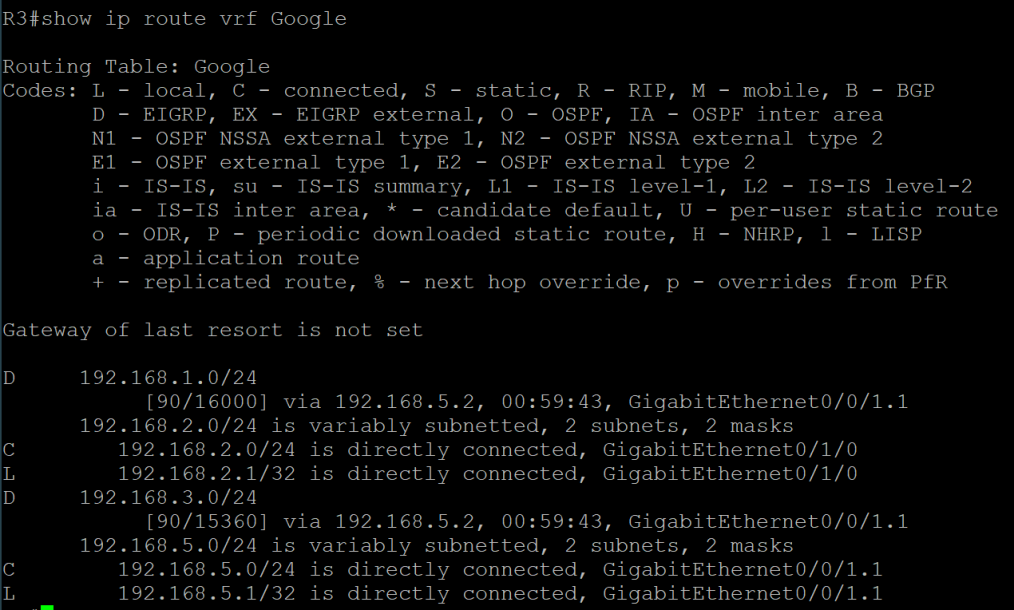
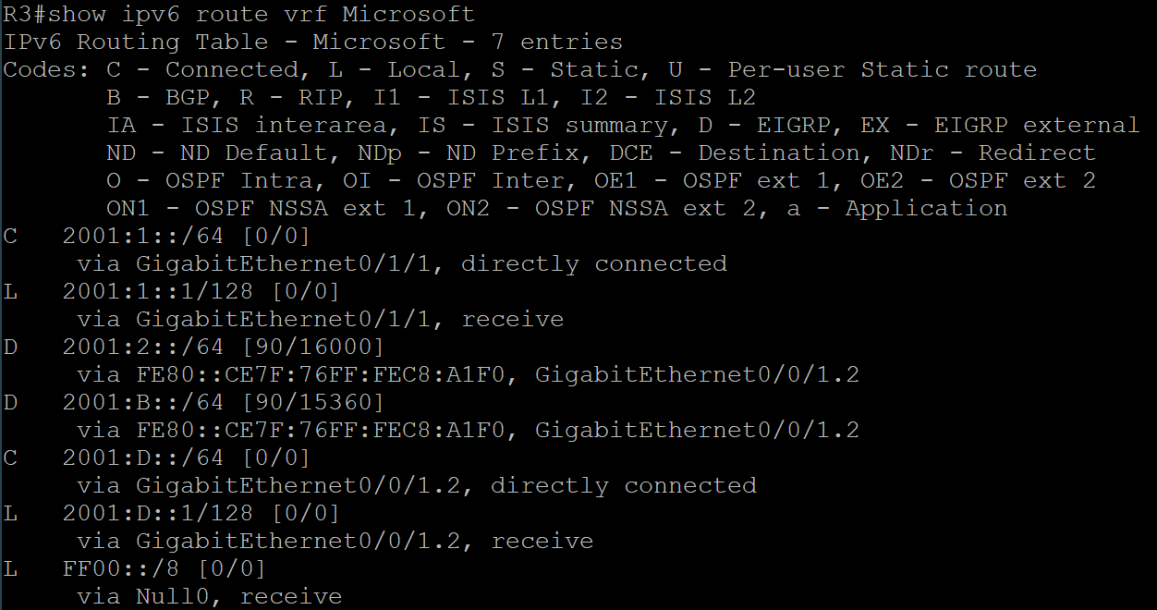
line vty 0 4

login

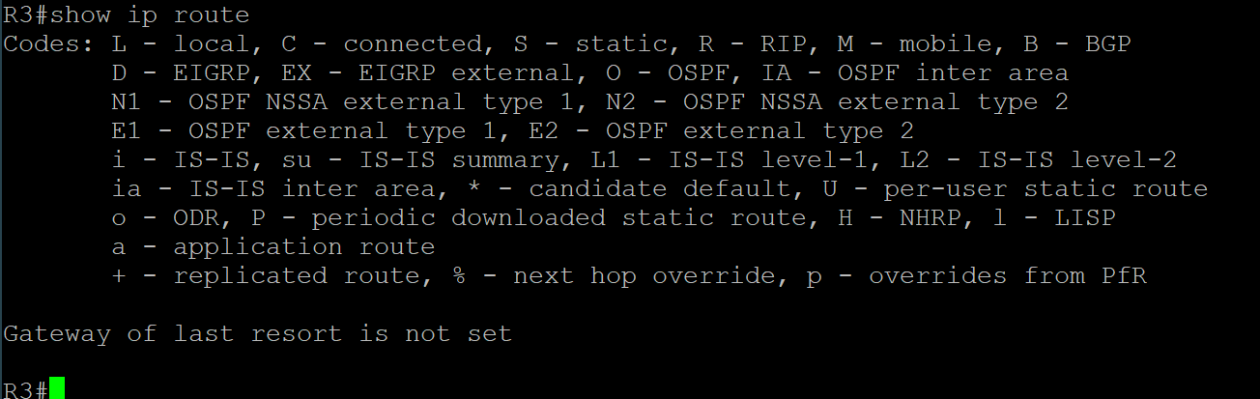
end

***Screenshots***

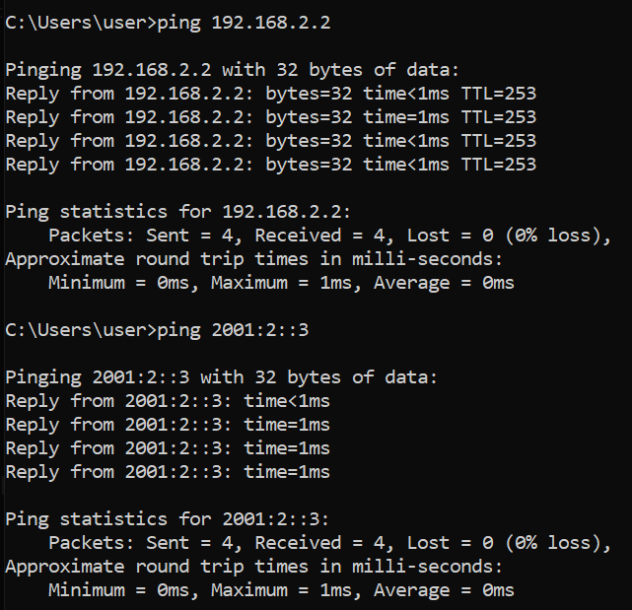
*Illustrated below is a picture of the IPv4 routing table for the networks currently in the VRF group Google. Note how it doesn’t show a route for 192.168.4.0 and 192.168.6.0. Which are exclusive to the VRF group Microsoft.*

*  
Shown below is the IPv6 routing table for the VRF group Microsoft, no routes to 2001:A::/64 or 2001:C::/64 are shown due to it not being included in the VRF group.  
*

*Indicated below is a picture of the routing table for Router 3 without specifying a VRF group, since all the networks connected and shared through EIGRP are participating in VRF, no results are available in this table.*

**

*Illustrated below is a picture of a ping from the PCs to the loopbacks to test connectivity across the network*

**

***Problems***

Right away the largest issue was with designing the lab topology in a way that would not require a cartographer to decipher. Originally, the topology wasn’t consistent in IPv4 and IPv6 addressing schemes and loopbacks were on both routers. Eventually, we decided that the point-to-point subnets would use letters in their IPv6 schemes and the others would use numbers. Once that part had been figured out, we initially wanted to use OSPF to distribute routes but found it much simpler to use EIGRP with VRF as more documentation was available that way. At first we also attempted to define VRF groups using a different command, however that would only allow for IPv4 VRF groups. Other than a few beginning setbacks, the rest of the lab went largely error free.

***Conclusion***

In summary, I now understand that VRF lite is a very useful way to virtually section off networks into separate routing tables without using multiple routers to achieve the same result. In addition, I comprehend how important it is to have a proper diagram for a network topology to avoid issues with explaining lab setups to other classmates.