**CCNP Lab 101: EIGRP và OSPF VRF**

1. **Mục tiêu bài lab:**

Trọng tâm của bài Lab này là tìm hiểu việc triển khai và xác minh Virtual Routing and Forwarding Lite (Định tuyến ảo và Chuyển tiếp Lite) trong phần mềm Cisco IOS bằng cách sử dụng EIGRP và OSPF. Các công nghệ bổ sung được thử nghiệm cũng bao gồm VRF-aware Network Address Translation (NAT).

1. **Lab Topology:**

A diagram of a network

Description automatically generated

1. **Đề bài:**

**Task 1:**

Cấu hình các VRFs sau trên các thiết bị trong mạng:

* R1: VRF Name - VRFONE: RD - 100:100
* R2: VRF Name - VRFTWO: RD - 200:200
* R3: VRF Name - VRFONE: RD - 100:100
* R3: VRF Name - VRFTWO: RD - 200:200
* R4: VRF Name - VRFONE: RD - 100:100
* R4: VRF Name - VRFTWO: RD - 200:200

Kiểm tra lại cấu hình sử dụng câu lệnh phù hợp.

**Task 2:**

Cấu hình EIGRP sử dụng AS 254 on R1 và gán interface Ethernet0/0 và LAN interfaces cho VRF VRFONE. Kiểm tra lại cấu hình sử dụng câu lệnh thích hợp.

**Task 3:**

Cấu hình OSPF trên R2 và gán interface Ethernet0/0 và LAN interfaces cho VRF VRFTWO. Kiểm tra cấu hình bằng câu lệnh phù hợp.

**Task 4:**

Cấu hình EIGRP và OSPF cho VRFONE và VRFTWO trên R3. Kiểm tra cấu hình sử dụng câu lệnh phù hợp.

**Task 5:**

Cấu hình EIGRP và OSPF cho VRFONE và VRFTWO trên R4. Kiểm tra cấu hình sử dụng câu lệnh phù hợp.

**Task 6:**

Cấu hình thêm 2 Loopback interfaces trên R4 như bên dưới:

Loopback 100: IP Address 100.4.4.4/32 - VRF: VRFONE Loopback 200: IP Address 100.4.4.4/32 - VRF: VRFTWO

**KHÔNG** quảng bá những Loopbacks này thông qua OSPF hay EIGRP. Tuy nhiên, hãy đảm bảo rằng R4 có thể ping tới mạng con 150.1.1.0/24 trong VRFONE từ Loopback100 interface và R4 có thể ping tới mạng con 150.2.2.0/24 trong VRFTWO từ Loopback200 interface.

**THỰC HÀNH LAB VRF**

**TASK 1:**

Trong phần mềm Cisco IOS, cấu hình VRF Lite là một nhiệm vụ đơn giản và dễ hiểu. Đầu tiên, tên VRF phải được xác định bằng lệnh cấu hình toàn cục **ip vrf [name]**. Tiếp theo, một trình phân biệt tuyến đường được gán cho VRF bằng cách sử dụng lệnh cấu hình **rd [route distinguisher]**. Bộ phân biệt tuyến đường được sử dụng để xác định VPN và được sử dụng để phân biệt nhiều tuyến VPN có tiền tố giống hệt nhau.

Lệnh cấu hình trên Router được thực hiện như sau:

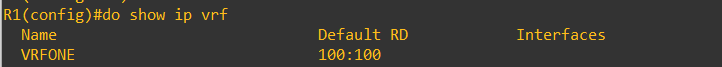
* Trên R1:

R1(config)#ip vrf VRFONE

R1(config-vrf)#rd 100:100

R1(config-vrf)#exit

Kết quả:



* Trên R2:

R2(config)#ip vrf VRFTWO

R2(config-vrf)#rd 200:200

R2(config-vrf)#exit

Kết quả:



* Trên R3:

R3(config)#ip vrf VRFONE

R3(config-vrf)#rd 100:100

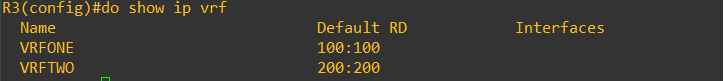
R3(config-vrf)#exit

R3(config)#ip vrf VRFTWO

R3(config-vrf)#rd 200:200

R3(config-vrf)#exit

Kết quả:



* Trên R4:

R4(config)#ip vrf VRFONE

R4(config-vrf)#rd 100:100

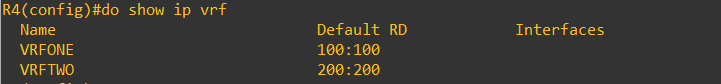
R4(config-vrf)#exit

R4(config)#ip vrf VRFTWO

R4(config-vrf)#rd 200:200

R4(config-vrf)#exit

Kết quả:



**TASK 2:**

Các yêu cầu cho task này rất đơn giản. Đối với EIGRP và RIPv2, cấu hình VRF được thực hiện bằng lệnh cấu hình router sử dụng **address-family ipv4 vrf [name].** Ngoài ra, đối với EIGRP, lệnh con cấu hình VRF **autonomous-system [number**] phải được sử dụng để chỉ định số hệ thống tự động. Định tuyến tĩnh cho mỗi VRF có thể được định cấu hình bằng cách sử dụng **ip route vrf [prefix] [mask] [interface | address]** lệnh cấu hình toàn cục.

Lệnh cấu hình được thực hiện như sau:

* Trên R1:

R1(config)#int e0/1

R1(config-if)#ip vrf forwarding VRFONE

R1(config-if)#ip add dhcp

R1(config-if)#exit

R1(config)#int e0/3

R1(config-if)#ip vrf forwarding VRFONE

R1(config-if)#ip add 10.0.0.5 255.255.255.252

R1(config-if)#exit

R1(config)#router eigrp 1

R1(config-router)#address-family ipv4 vrf VRFONE

R1(config-router-af)#network 10.215.27.121 0.0.0.0

R1(config-router-af)#network 10.0.0.5 0.0.0.0

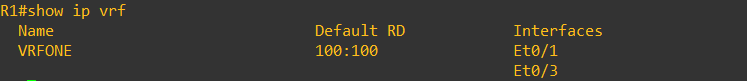
R1(config-router-af)#no auto-summary

R1(config-router-af)#autonomous-system 254

R1(config-router-af)#exit

R1(config-router)#exit

Thực hiện câu lệnh **show ip vrf :**

****

Thực hiện câu lệnh **show ip eigrp vrf VRFONE interfaces** và kết quả là:

A screenshot of a computer screen

Description automatically generated

**TASK 3:**

Các yêu cầu cho task này rất đơn giản. Trong phần mềm Cisco IOS, VRF Lite sử dụng OSPF được định cấu hình bằng lệnh cấu hình toàn cục **router ospf [process ID] vrf [name]**, theo sau là lệnh **capability vrf-lite**. Phần còn lại của cấu hình, ví dụ: các câu lệnh mạng, được thực hiện theo cách tương tự như khi VRF Lite không được sử dụng. Task này được thực hiện như sau:

Lệnh cấu hình được thực hiện như sau:

* Trên R2:

R2(config)#int e0/1

R2(config-if)#ip vrf forwarding VRFTWO

R2(config-if)#ip add dhcp

R2(config-if)#exit

R2(config)#int e0/0

R2(config-if)#ip vrf forwarding VRFTWO

R2(config-if)#ip add 10.0.0.9 255.255.255.252

R2(config-if)#exit

R2(config)#router ospf 2 vrf VRFTWO

R2(config-router)#capability vrf-lite

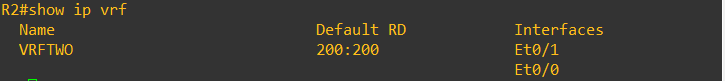
R2(config-router)#router-id 2.2.2.2

R2(config-router)#net 10.215.27.122 0.0.0.0 area 0

R2(config-router)#network 10.0.0.9 0.0.0.0 area 0

R2(config-router)#exit

Thực hiện câu lệnh **show ip vrf:**

****

Thực hiện câu lệnh **show ip ospf interface brief** và kết quả là:

A screen shot of a computer

Description automatically generated

**TASK 4:**

Các yêu cầu của task này rất đơn giản. Theo cùng một logic được sử dụng trong các tác vụ trước, tác vụ này được hoàn thành trên R3 như sau:

R3(config)#int e0/0

R3(config-if)#ip vrf forwarding VRFTWO

R3(config-if)#ip add 10.0.0.10 255.255.255.252

R3(config-if)#exit

R3(config)#int e0/3

R3(config-if)#ip vrf forwarding VRFONE

R3(config-if)#ip add 10.0.0.6 255.255.255.252

R3(config-if)#exit

R3(config)#int e0/2

R3(config-if)#ip vrf forwarding VRFONE

R3(config-if)#ip add 10.0.0.13 255.255.255.252

R3(config-if)#exit

R3(config)#int e0/1

R3(config-if)#ip vrf forwarding VRFTWO

R3(config-if)#ip add 10.0.0.17 255.255.255.252

R3(config-if)#exit

R3(config)#exit

Thực hiện câu lệnh **show ip vrf:**

A screen shot of a computer

Description automatically generated

Tiếp thực hiện thực hiện cấu hình EIGRP và OSPF cho R3:

R3(config)#router eigrp 1

R3(config-router)#address-family ipv4 vrf VRFONE

R3(config-router-af)#no auto-summary

R3(config-router-af)#network 10.0.0.6 0.0.0.0

R3(config-router-af)#network 10.0.0.13 0.0.0.0

R3(config-router-af)#exit

R3(config-router-af)#exit

R3(config-router)#exit

R3(config)#router ospf 3 vrf VRFTWO

R3(config-router)#capability vrf-lite

R3(config-router)#router-id 3.3.3.3

R3(config-router)#network 10.0.0.10 0.0.0.0 area 0

R3(config-router)#network 10.0.0.17 0.0.0.0 area 0

R3(config-router)#exit

R3(config)#exit

Thực hiện **show ip eigrp vrf VRFONE interfaces:**

A screenshot of a computer

Description automatically generated

Thực hiện **show ip eigrp vrf VRFONE neighbors:**

**A black background with yellow text

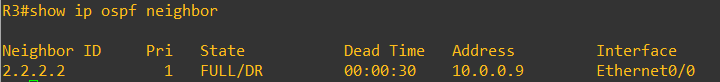
Description automatically generated**

Thực hiện **show ip ospf interface brief:**

**A screen shot of a number

Description automatically generated**

Thực hiện **show ip ospf neighbor:**



**TASK 5:**

Theo cùng một logic được áp dụng trong các task trước, tác vụ này được hoàn thành trên R4 như sau:

R4(config)#int e0/2

R4(config-if)#ip vrf forwarding VRFONE

R4(config-if)#ip add 10.0.0.14 255.255.255.252

R4(config-if)#exit

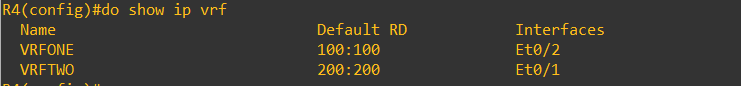
R4(config)#int e0/1

R4(config-if)#ip vrf forwarding VRFTWO

R4(config-if)#ip add 10.0.0.18 255.255.255.252

R4(config-if)#exit

Thực hiện câu lệnh **show ip vrf:**

****

Thực hiện **show ip eigrp vrf VRFONE neighbors:**

A black screen with yellow text

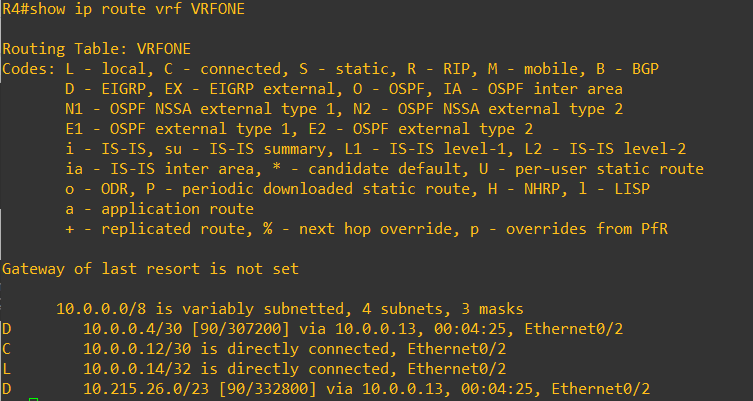
Description automatically generated

Thực hiện **show ip ospf neighbor:**

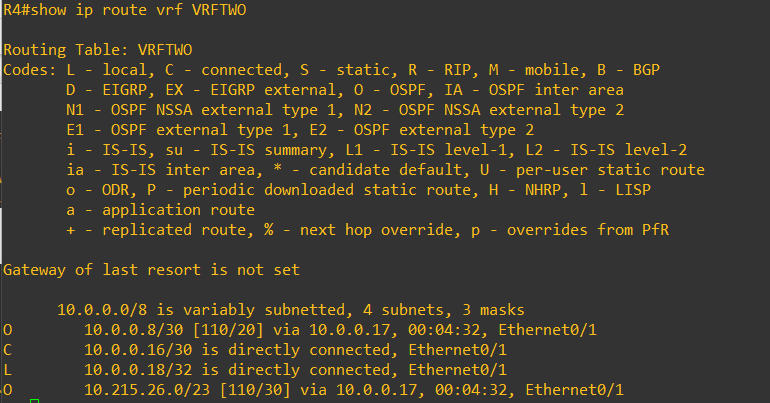
A screen shot of a computer

Description automatically generated

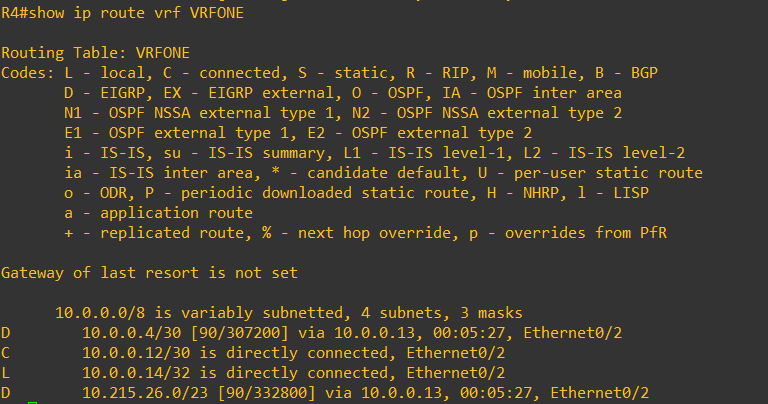
Thực hiện **show ip route vrf VRFONE**:



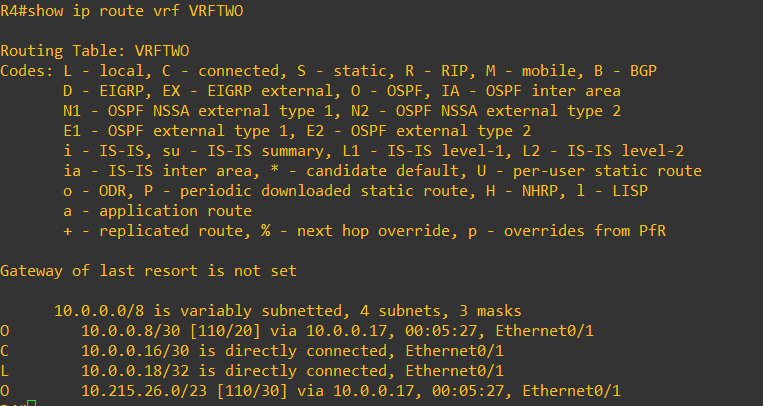
Thực hiện **show ip route vrf VRFTWO**:



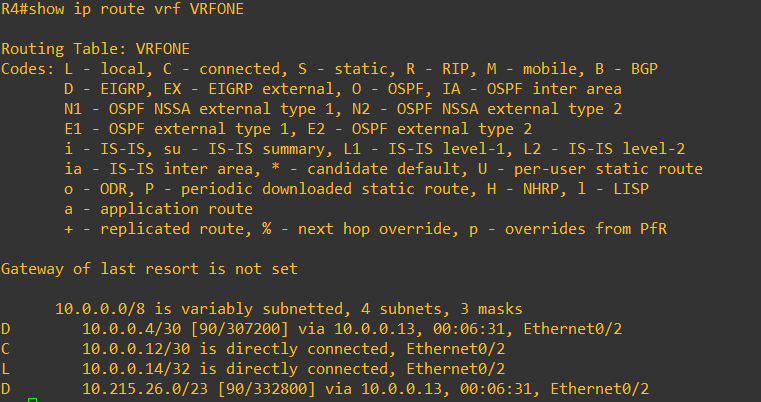
Thực hiện **show ip route vrf VRFONE:**



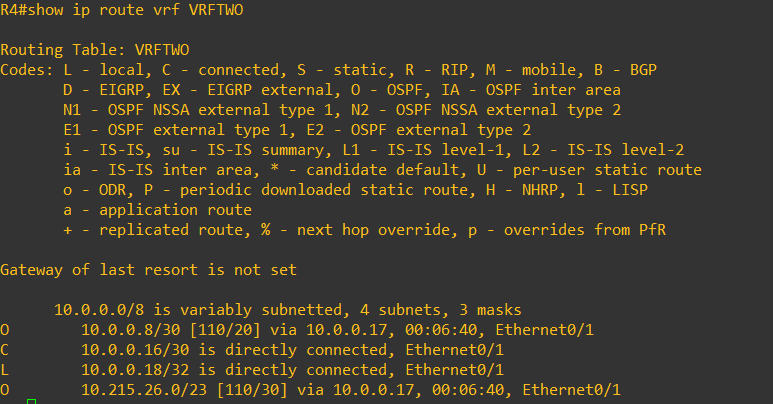
Thực hiện **show ip route vrf VRFTWO:**



Thực hiện **show ip route vrf VRFONE**:



Thực hiện **show ip route vrf VRFTWO**:



**TASK 6:**

Chúng tôi đã đưa VRF Lite vào mạng không có nghĩa là các nguyên tắc cơ bản hoặc cốt lõi không được áp dụng. Nếu VRF Lite không được sử dụng, chúng tôi sẽ sử dụng NAT để hoàn thành task này. May mắn thay, Cisco IOS NAT là VRF-aware. Do đó, task này được hoàn thành trên R4 như sau:

R4(config)#interface loopback 100

R4(config-if)#ip vrf forwarding VRFONE

R4(config-if)#ip address 100.4.4.4 255.255.255.255

R4(config-if)#ip nat inside

R4(config-if)#exit

R4(config)#interface loopback 200

R4(config-if)#ip vrf forwarding VRFTWO

R4(config-if)#ip address 100.4.4.4 255.255.255.255

R4(config-if)#ip nat inside

R4(config-if)#exit

R4(config)#interface e0/1

R4(config-if)#ip nat outside

R4(config-if)#exit

R4(config)#interface e0/2

R4(config-if)#ip nat outside

R4(config-if)#exit

R4(config)#access-list 100 permit ip host 100.4.4.4 any

R4(config)#ip nat inside source list 100 interface ethernet 0/2 vrf VRFONE overload

R4(config)#ip nat inside source list 100 interface ethernet 0/1 vrf VRFTWO overload

Thực hiện câu lệnh **ping vrf VRFONE 10.215.27.121 source loopback 100:**

A screenshot of a computer

Description automatically generated

Thực hiện câu lệnh **show ip nat translations vrf VRFONE verbose:**

A screenshot of a computer

Description automatically generated

Thực hiện **ping vrf VRFTWO 10.215.27.122 source loopback 200:**

**A screen shot of a computer

Description automatically generated**

Thực hiện **show ip nat translations vrf VRFTWO verbose:**

A screen shot of a computer

Description automatically generated