

# Red virtual fácil de la configuración con el EIGRP Mode Nombrado

## Contenido

[Introducción](#)

[prerrequisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Configurar](#)

[Diagrama de la red](#)

[Configuraciones](#)

[Verificación](#)

[Inheritance con el modo Nombrado del EIGRP](#)

[Replicación de la ruta con el modo del nombre del EIGRP](#)

[Contexto de ruteo](#)

[Traceroute aumentado](#)

[Conclusión](#)

[Referencias](#)

## Introducción

El propósito de este documento es demostrar la configuración de EVN (red virtual fácil) usando el (Enhanced Interior Gateway Routing Protocol) del EIGRP nombrado modo. Es un suplemento al [documento sobre configuración fácil de la red virtual](#), que demuestra el uso de OSPF (trayecto más corto abierto primero), así como a otros temas más complejos como las listas del trunk VNET y la replicación de la ruta. El EVN VNET fue pensado para que a los operadores tengan una opción más fácil que MPLS (Multi Protocol Label Switching) VPN (Virtual Private Network) o VRF-lite (ruteo virtual y expedición) para desplegar los VRF múltiples. EVN VNET utiliza un concepto de configuración reproducida para que los Routing Protocol y la interfaz de tronco VNET quiten la carga del operador y salven algunas de las tareas repetidores. Resolviendo problemas el EIGRP, la encaminamiento o el CEF (Cisco Express Forwarding) está fuera del alcance de este documento y a menos que sea conocido usted pueda seguir los procedimientos de Troubleshooting normales.

## Prerrequisitos

### Requisitos

Cisco recomienda que usted tiene conocimiento básico del EIGRP.

Esta característica es available en pocas versiones después de la versión de IOS 15.2. Para verificar si el EIGRP nombrado modo con EVN VNET se soporta, marque la salida de los **plug-in del eigrp del IP de la demostración**. Si es fácil la versión 1.00.00 o posterior de la red virtual está presente, entonces sus soportes de versión esta característica.

```
R1#show eigrp plugins
EIGRP feature plugins:::
eigrp-release : 21.00.00 : Portable EIGRP Release
: 1.00.10 : Source Component Release(rel21)
parser : 2.02.00 : EIGRP Parser Support
igrp2 : 2.00.00 : Reliable Transport/Dual Database
bfd : 2.00.00 : BFD Platform Support
mtr : 1.00.01 : Multi-Topology Routing(MTR)
eigrp-pfr : 1.00.01 : Performance Routing Support
EVN/vNets : 1.00.00 : Easy Virtual Network (EVN/vNets)
ipv4-af : 2.01.01 : Routing Protocol Support
ipv4-sf : 1.02.00 : Service Distribution Support
vNets-parse : 1.00.00 : EIGRP vNets Parse Support
ipv6-af : 2.01.01 : Routing Protocol Support
ipv6-sf : 2.01.00 : Service Distribution Support
snmp-agent : 2.00.00 : SNMP/SNMPv2 Agent Support
```

Nota: El EIGRP nombrado modo con EVN VNETs no se soporta en 15.1SY. En esta versión usted debe utilizar la configuración EIGRP clásica del modo que se demuestra ya en la documentación disponible.

El BFD (detección bidireccional de la expedición) se soporta actualmente solamente en VNET global y no funcionará en ningunos subinterfaces Nombrados VNET en el trunk VNET.

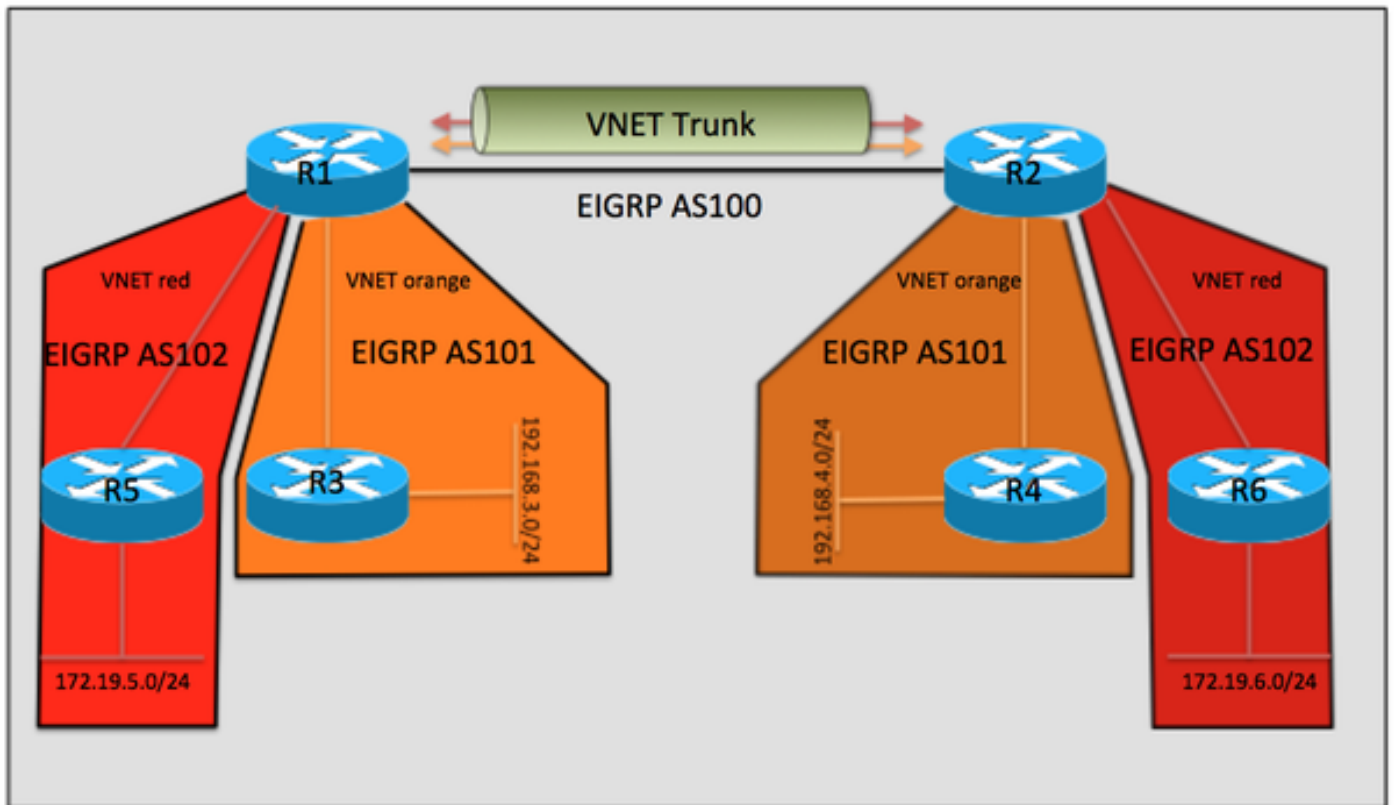
No se aconseja para utilizar el valor por defecto de la AF-interfaz al usar el EIGRP nombró el modo con EVN VNETs debido a la herencia imprevisible posible.

## Componentes Utilizados

La información en este documento fue creada de los dispositivos en una versión deL Cisco IOS corriente 15.6(1)S2 del ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración verificada (predeterminada). Si la red está funcionando, asegúrese de haber comprendido el impacto que puede tener cualquier comando.

## Configurar

### Diagrama de la red



## Configuraciones

Las configuraciones de R3, de R4, de R5 y de R6 son todo documento similar, y por lo tanto a la izquierda de los. Se configuran simplemente para formar a un vecino EIGRP con el r1 o el r2, y no son conscientes del EVN VNET usado entre el r1 y el r2.

### Configuración pertinente del r1

```
vrf definition orange
 vnet tag 101
 !
 address-family ipv4
 exit-address-family
 !
vrf definition red
 vnet tag 102
 !
 address-family ipv4
 exit-address-family
 !
interface Ethernet0/0
 vnet trunk
 ip address 10.12.12.1 255.255.255.0
 !
interface Ethernet1/0
 vrf forwarding orange
 ip address 192.168.13.1 255.255.255.0
 !
interface Ethernet2/0
 vrf forwarding red
 ip address 192.168.15.1 255.255.255.0
 !
 !
router eigrp named
 !
```

```

address-family ipv4 unicast autonomous-system 100
!
af-interface Ethernet0/0
authentication mode hmac-sha-256 cisco
exit-af-interface
!
topology base
exit-af-topology
network 10.0.0.0
exit-address-family
!
address-family ipv4 unicast vrf orange autonomous-system 101
!
af-interface Ethernet1/0
authentication mode hmac-sha-256 cisco
exit-af-interface
!
topology base
exit-af-topology
network 10.0.0.0
network 192.168.13.0
exit-address-family
!
address-family ipv4 unicast vrf red autonomous-system 102
!
topology base
exit-af-topology
network 10.0.0.0
network 192.168.15.0
exit-address-family

```

## Configuración pertinente del r2

```

vrf definition orange
vnet tag 101
!
address-family ipv4
exit-address-family
!
vrf definition red
vnet tag 102
!
address-family ipv4
exit-address-family
!
interface Ethernet0/0
vnet trunk
ip address 10.12.12.2 255.255.255.0
!
interface Ethernet1/0
vrf forwarding orange
ip address 192.168.24.2 255.255.255.0
!
interface Ethernet2/0
vrf forwarding red
ip address 192.168.26.2 255.255.255.0
!
!
router eigrp named
!
address-family ipv4 unicast autonomous-system 100
!
af-interface Ethernet0/0
authentication mode hmac-sha-256 cisco

```

```

exit-af-interface
!
topology base
exit-af-topology
  network 10.0.0.0
exit-address-family
!
  address-family ipv4 unicast vrf orange autonomous-system 101
!
af-interface Ethernet1/0
authentication mode hmac-sha-256 cisco
exit-af-interface
!
topology base
exit-af-topology
network 10.0.0.0
  network 192.168.24.0
exit-address-family
!
  address-family ipv4 unicast vrf red autonomous-system 102
!
topology base
exit-af-topology
network 10.0.0.0
  network 192.168.26.0
exit-address-family

```

## Verificación

Una de las ventajas de la red virtual fácil es la simplicidad de la configuración. Esto es alcanzada automáticamente configurando los trunks VNET para cada etiqueta VNET. Comparando EVN con VRF-lite, cada sub-interfaz necesitaría ser configurada manualmente. El Ethernet0/0 es el trunk VNET que conecta el r1 y el r2, y una sub-interfaz VNET es creada automáticamente para que cada VNET cumpla los requisitos de la separación del tráfico para EVN añadiendo las tramas al final del fichero con una etiqueta del dot1q VNET. Estos subinterfaces no son visibles en la salida de la ejecutar-configuración de la demostración, no obstante pueden ser considerados con los derivar-config de la demostración.

```

R1#show derived-config | sec Ethernet0/0
interface Ethernet0/0
  vnet trunk
ip address 10.12.12.1 255.255.255.0
no ip redirects
no ip proxy-arp
interface Ethernet0/0.101
  description Subinterface for VNET orange
  encapsulation dot1Q 101
  vrf forwarding orange
  ip address 10.12.12.1 255.255.255.0
  no ip proxy-arp
interface Ethernet0/0.102
  description Subinterface for VNET red
  encapsulation dot1Q 102
  vrf forwarding red
  ip address 10.12.12.1 255.255.255.0
  no ip proxy-arp

```

Semejantemente, usted puede ver que la configuración EIGRP también está creada automáticamente:

```

R1#show derived-config | sec router eigrp

```

```

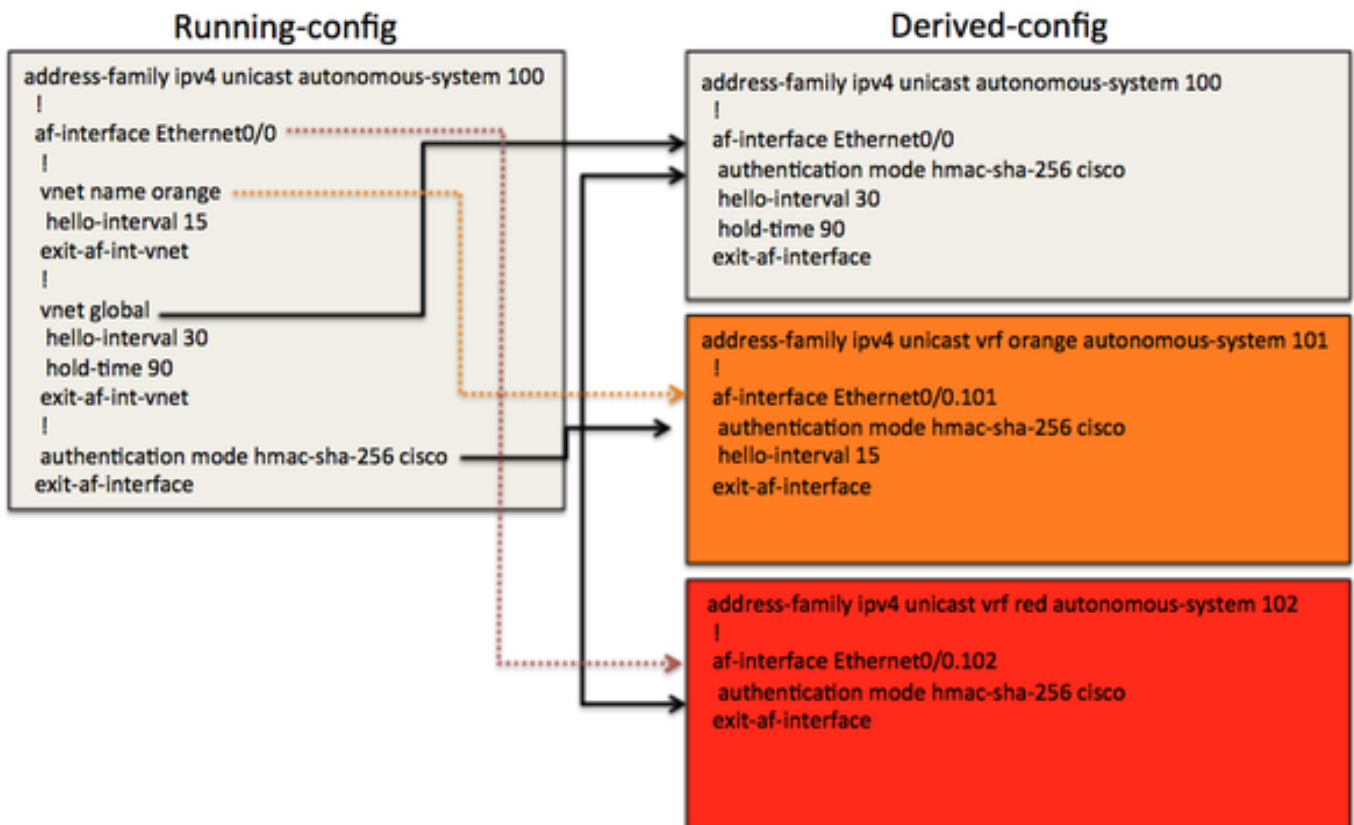
router eigrp named
!
address-family ipv4 unicast autonomous-system 100
!
af-interface Ethernet0/0
authentication mode hmac-sha-256 cisco
exit-af-interface
!
topology base
exit-af-topology
network 10.0.0.0
exit-address-family
!
address-family ipv4 unicast vrf orange autonomous-system 101
!
af-interface Ethernet0/0.101
authentication mode hmac-sha-256 cisco
exit-af-interface
!
af-interface Ethernet1/0
authentication mode hmac-sha-256 cisco
exit-af-interface
!
topology base
exit-af-topology
network 10.0.0.0
network 192.168.13.0
exit-address-family
!
address-family ipv4 unicast vrf red autonomous-system 102
!
af-interface Ethernet0/0.102
authentication mode hmac-sha-256 cisco
exit-af-interface
!
topology base
exit-af-topology
network 10.0.0.0
network 192.168.15.0
exit-address-family
R1#

```

Una observación interesante en la salida antedicha es la herencia de la AF-interfaz para los subinterfaces VNET de la AF-interfaz ethernet0/0 en el Autonomous System global 100 del vrf. La sección de siguiente explica esto más en detalle:

## Inheritance con el modo Nombrado del EIGRP

La figura abajo será utilizada para ayudar a visualizar las reglas de la herencia al usar el EIGRP nombró el modo con EVN VNETs.



En el ejemplo anterior hay una AF-interfaz ethernet0/0 del trunk VNET, de la cual los subinterfaces VNET recibirán su configuración derivada. La configuración de algunos valores no predeterminados tales como intervalo de saludo, tiempo en espera y autenticación se ha hecho para demostrar la herencia. Usted también notará el VNET sub-MODE bajo AF-interfaz en el proceso EIGRP global. Esto es una manera de controlar qué opciones de configuración se reproducen a la AF-interfaz dinámicamente creada para cada VNET dentro de su configuración del vrf del EIGRP.

Por ejemplo el config derivado para Eth0/0 en la tabla de Global Routing se hereda del vnet global (el intervalo de saludo 30, el tiempo en espera 90). Configuran al modo de autenticación hmac-sha-256 para Eth0/0 directamente en esta AF-interfaz en los ejecutar-config, y la salida derivada de los config muestra que Eth0/0 ha heredado el comando. Puesto que configuran al modo de autenticación en la AF-interfaz del trunk VNET, es heredada por todas las interfaces VNET.

Para la naranja del vrf, la naranja VNET se ha configurado con un intervalo de saludo de 15 en los ejecutar-config. En los config derivados que usted puede ver para la naranja VRF en el Autonomous System 101, el intervalo de saludo de 15 fue tomado del VNET sub-MODE bajo AF-interfaz eth0/0, en el proceso global. El tiempo en espera no fue modificado y fue reproducido de la AF-interfaz eth0/0 que está utilizando el valor predeterminado.

El rojo VNET no tiene ninguna diferencia en la configuración de la AF-interfaz Eth0/0, así que hereda los valores de temporizador predeterminado así como al modo de autenticación.

Estas opciones de configuración permiten la flexibilidad para que el operador utilice diversos parámetros para cada sub-interfaz del trunk VNET. Por ejemplo, diversos valores del temporizador, modos de autenticación o interfaz pasiva. Para resumir las reglas de la herencia, todo el VNETs heredará la configuración de la AF-interfaz del trunk VNET. La configuración específica VNET en VNET sub-MODE también será heredada por los subinterfaces del trunk VNET, y toma la prioridad sobre los parámetros de la AF-interfaz.

Abajo está una cierta salida adicional para verificar la herencia de la configuración:

```
R1#show eigrp address-family ipv4 interface detail e0/0
EIGRP-IPv4 VR(named) Address-Family Interfaces for AS(100)
Xmit Queue PeerQ Mean Pacing Time Multicast Pending
Interface Peers Un/Reliable Un/Reliable SRTT Un/Reliable Flow Timer Routes
Et0/0 1 0/0 0/0 6 0/2 50 0
Hello-interval is 30, Hold-time is 90
Split-horizon is enabled
Next xmit serial <none>
Packetized sent/expedited: 3/1
Hello's sent/expedited: 2959/3
Un/reliable mcasts: 0/4 Un/reliable ucasts: 5/5
Mcast exceptions: 0 CR packets: 0 ACKs suppressed: 0
Retransmissions sent: 3 Out-of-sequence rcvd: 1
Topology-ids on interface - 0
Authentication mode is HMAC-SHA-256, key-chain is not set
Topologies advertised on this interface: base
Topologies not advertised on this interface:
```

```
R1#show eigrp address-family ipv4 vrf orange interface detail e0/0.101
EIGRP-IPv4 VR(named) Address-Family Interfaces for AS(101)
VRF(orange)
Xmit Queue PeerQ Mean Pacing Time Multicast Pending
Interface Peers Un/Reliable Un/Reliable SRTT Un/Reliable Flow Timer Routes
Et0/0.101 1 0/0 0/0 5 0/2 50 0
Hello-interval is 15, Hold-time is 15
Split-horizon is enabled
Next xmit serial <none>
Packetized sent/expedited: 4/1
Hello's sent/expedited: 2371/3
Un/reliable mcasts: 0/4 Un/reliable ucasts: 6/5
Mcast exceptions: 0 CR packets: 0 ACKs suppressed: 0
Retransmissions sent: 3 Out-of-sequence rcvd: 1
Topology-ids on interface - 0
Authentication mode is HMAC-SHA-256, key-chain is not set
Topologies advertised on this interface: base
Topologies not advertised on this interface:
```

```
R1#show eigrp address-family ipv4 vrf red interface detail e0/0.102
EIGRP-IPv4 VR(named) Address-Family Interfaces for AS(102)
VRF(red)
Xmit Queue PeerQ Mean Pacing Time Multicast Pending
Interface Peers Un/Reliable Un/Reliable SRTT Un/Reliable Flow Timer Routes
Et0/0.102 1 0/0 0/0 4 0/2 50 0
Hello-interval is 5, Hold-time is 15
Split-horizon is enabled
Next xmit serial <none>
Packetized sent/expedited: 6/1
Hello's sent/expedited: 2676/3
Un/reliable mcasts: 0/6 Un/reliable ucasts: 7/5
Mcast exceptions: 0 CR packets: 0 ACKs suppressed: 0
Retransmissions sent: 3 Out-of-sequence rcvd: 1
Topology-ids on interface - 0
Authentication mode is HMAC-SHA-256, key-chain is not set
Topologies advertised on this interface: base
Topologies not advertised on this interface:
```

## Replicación de la ruta con el modo del nombre del EIGRP

Una de las ventajas de EVN es la capacidad de replicar las rutas entre VNETs. Por ejemplo el R4



en el rojo VRF puede necesitar alcanzar un servicio en 192.168.13.0/24 que sea naranja de la parte de VRF. Esto se puede alcanzar usando la configuración abajo.

```
R2#show run
vrf definition orange
vnet tag 101
!
address-family ipv4
exit-address-family
!
vrf definition red
vnet tag 102
!
address-family ipv4
route-replicate from vrf orange unicast eigrp 101 route-map filter
exit-address-family
!
<output removed>
!
ip prefix-list filter seq 5 permit 192.168.13.0/24
!
route-map filter permit 10
  match ip address prefix-list filter
!
```

Ahora el prefijo 192.168.13.0/24 está en el rojo VRF, no obstante el ping no está trabajando porque la dirección de origen no es ruta replicada en la naranja VNET.

```
R2#show ip route vrf red
```

```
Routing Table: red
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
a - application route
+ - replicated route, % - next hop override, p - overrides from Pfr

Gateway of last resort is not set

10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
D 10.5.5.5/32 [90/1536640] via 10.12.12.1, 03:48:46, Ethernet0/0.102
D 10.6.6.6/32 [90/1024640] via 192.168.26.6, 03:48:37, Ethernet2/0
C 10.12.12.0/24 is directly connected, Ethernet0/0.102
L 10.12.12.2/32 is directly connected, Ethernet0/0.102
D + 192.168.13.0/24
[90/1536000] via 10.12.12.1 (orange), 03:48:46, Ethernet0/0.101
D 192.168.15.0/24 [90/1536000] via 10.12.12.1, 03:48:46, Ethernet0/0.102
192.168.26.0/24 is variably subnetted, 2 subnets, 2 masks
C 192.168.26.0/24 is directly connected, Ethernet2/0
L 192.168.26.2/32 is directly connected, Ethernet2/0
R2#
R2#
R2#ping vrf red 192.168.13.1 source e2/0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.13.1, timeout is 2 seconds:
Packet sent with a source address of 192.168.26.2
.....
Success rate is 0 percent (0/5)
```

Después de todo rutas replicadas del rojo VRF a la naranja VRF en el r1, usando la configuración similar:

```
R2#show ip route vrf red
```

```
Routing Table: red
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
a - application route
+ - replicated route, % - next hop override, p - overrides from PFR
```

```
Gateway of last resort is not set
```

```
10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
D 10.5.5.5/32 [90/1536640] via 10.12.12.1, 03:48:46, Ethernet0/0.102
D 10.6.6.6/32 [90/1024640] via 192.168.26.6, 03:48:37, Ethernet2/0
C 10.12.12.0/24 is directly connected, Ethernet0/0.102
L 10.12.12.2/32 is directly connected, Ethernet0/0.102
D + 192.168.13.0/24
[90/1536000] via 10.12.12.1 (orange), 03:48:46, Ethernet0/0.101
D 192.168.15.0/24 [90/1536000] via 10.12.12.1, 03:48:46, Ethernet0/0.102
192.168.26.0/24 is variably subnetted, 2 subnets, 2 masks
C 192.168.26.0/24 is directly connected, Ethernet2/0
L 192.168.26.2/32 is directly connected, Ethernet2/0
```

```
R2#
R2#
R2#ping vrf red 192.168.13.1 source e2/0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.13.1, timeout is 2 seconds:
Packet sent with a source address of 192.168.26.2
.....
Success rate is 0 percent (0/5)
```

Nota: Usted puede ruta-réplica conectada, BGP, EIGRP, etc. Refiera por favor a las referencias por más ejemplos.

## Contexto de ruteo

Otra buena característica con EVN es el concepto de contexto de ruteo. Esto permite que usted ejecute los comandos dentro del rojo VRF, sin tener que incluir el "rojo del vrf" en cada CLI. Por ejemplo, lo mismo hacen ping como arriba usando el contexto de ruteo se muestra abajo.

```
R2#routing-context vrf red
R2%red#ping 192.168.13.1 source e2/0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.13.1, timeout is 2 seconds:
Packet sent with a source address of 192.168.26.2
!!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
R2%red#
```

## Traceroute aumentado

La salida del comando traceroute también visualizará los nombres VRF VNET, que es útil para resolver problemas, especialmente si la replicación de la ruta está implicada.

```
R6#traceroute 192.168.13.3
Type escape sequence to abort.
Tracing the route to 192.168.13.3
VRF info: (vrf in name/id, vrf out name/id)
 1 192.168.26.2 (red,orange/101) 1 msec 0 msec 0 msec
 2 10.12.12.1 (orange/101,orange) 2 msec 1 msec 1 msec
 3 192.168.13.3 0 msec * 1 msec
```

La misma traza del r2

```
R2#trace vrf red 192.168.13.3 source 192.168.26.2
Type escape sequence to abort.
Tracing the route to 192.168.13.3
VRF info: (vrf in name/id, vrf out name/id)
 1 10.12.12.1 (orange/101,orange) 1 msec 1 msec 0 msec
 2 192.168.13.3 1 msec * 1 msec
```

En esta salida usted puede ver que del r2, el Next-Hop en la naranja VRF está tomado directamente para alcanzar 192.168.13.0/24.

## Conclusión

La configuración EVN VNET con el EIGRP nombrado modo proporciona una manera para que los clientes desplieguen un entorno de red virtualizado, y quita algo de la complejidad asociada al MPLS VPN tradicional, o VRF-lite. La comprensión de las reglas de la herencia es dominante con éxito a desplegar esta característica y asegurar la red está actuando según lo previsto.

## Referencias

Whitepaper fácil de las redes virtuales

[http://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/layer-3-vpns-l3vpn/whitepaper\\_c11-638769.html](http://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/layer-3-vpns-l3vpn/whitepaper_c11-638769.html)

Guía de configuración

<http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/evn/configuration/xe-3s/evn-xe-3s-book/evn-overview.html>