Ejemplo de Configuración de EIGRP en SVTI, DVTI e IKEv2 FlexVPN con el Comando "IP[v6] Unnumbered"

Contenido

Introducción
Prerequisites
Requirements
Componentes Utilizados
EIGRP en un segmento Ethernet con diferentes subredes
EIGRP en segmento SVTI con diferentes subredes
Utilice el comando IP Unnumbered
EIGRP en SVTI para el segmento DVTI con diferentes subredes
EIGRP en IKEv2 Flex VPN con diferentes subredes
Modo de configuración para routing
EIGRP IPV6 en segmento SVTI con diferentes subredes
EIGRP IPV6 en IKEv2 Flex VPN con diferentes subredes
Verificación
Troubleshoot
Advertencias conocidas
Summary
Información Relacionada

Introducción

Este documento describe cómo configurar el protocolo de routing de gateway interior mejorado (EIGRP) en una serie de escenarios más comunes en Cisco IOS®. Para aceptar una adyacencia de vecino EIGRP, Cisco IOS debe obtener el paquete HELLO EIGRP de una dirección IP dentro de la misma subred. Es posible inhabilitar esa verificación con el comando **ip unnumbered**.

La primera parte del artículo presenta una falla de EIGRP cuando recibe un paquete que no está en la misma subred.

Otro ejemplo demuestra el uso del comando **ip unnumbered** que inhabilita esa verificación y permite que EIGRP forme una adyacencia entre peers que pertenecen a diferentes subredes.

En este artículo también se presenta una implementación de FlexVPN Hub y Spoke con una dirección IP enviada desde el servidor. Para este escenario, la verificación de subredes se inhabilita para el comando **ip address negotiated** y también para el comando **ip unnumbered**. El comando **ip unnumbered** se utiliza principalmente para las interfaces de tipo punto a punto (P2P),

lo que hace que FlexVPN sea un ajuste perfecto ya que se basa en una arquitectura P2P.

Por último, se presenta un escenario IPv6 junto con las diferencias tanto para las interfaces de túnel virtual estáticas (SVTI) como para las interfaces de túnel virtual dinámico (DVTI). Hay ligeros cambios en el comportamiento al comparar los escenarios de IPv6 con IPv4.

Además, se presentan los cambios entre las versiones 15.1 y 15.3 del IOS de Cisco (Id. de bug Cisco CSCtx45062).

El comando **ip unnumbered** siempre es necesario para DVTI. Esto se debe a que las direcciones IP configuradas estáticamente en una interfaz de plantilla virtual nunca se clonan en una interfaz de acceso virtual. Además, una interfaz sin una dirección IP configurada no puede establecer ninguna adyacencia de protocolo de ruteo dinámico. El comando **ip unnumbered** no es necesario para SVTI, pero sin esa subred, la verificación se realiza cuando se establece la adyacencia del protocolo de ruteo dinámico. Además, el comando **ipv6 unnumbered** no es necesario para los escenarios IPV6 debido a las direcciones locales de link que se utilizan para generar adyacencias EIGRP.

Prerequisites

Requirements

Cisco recomienda que tenga conocimientos básicos sobre estos temas:

- Configuración de VPN en Cisco IOS
- Configuración de FlexVPN en Cisco IOS

Componentes Utilizados

La información de este documento se basa en la versión 15.3T del IOS de Cisco.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

EIGRP en un segmento Ethernet con diferentes subredes

Topología: Router 1 (R1) (e0/0: 10.0.0.1/24)—(e0/1: 10.0.1.2/24) Router 2 (R2)

R1:

```
interface Ethernet0/0
 ip address 10.0.0.1 255.255.255.0

router eigrp 100
network 10.0.0.1 0.0.0.0
```

R2:

```
interface Ethernet0/0
```

```
ip address 10.0.1.2 255.255.255.0

router eigrp 100
network 10.0.1.2 0.0.0.0
```

R1 muestra:

```
*Mar 3 16:39:34.873: EIGRP: Received HELLO on Ethernet0/0 nbr 10.0.1.2
*Mar 3 16:39:34.873:   AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0
*Mar 3 16:39:34.873: EIGRP-IPv4(100): Neighbor 10.0.1.2 not on common subnet
for Ethernet0/0
```

Cisco IOS no forma una adyacencia, lo que se espera. Para obtener más información sobre esto, refiérase a ¿Qué Significan los Mensajes EIGRP "No en una Subred Común"? artículo.

EIGRP en segmento SVTI con diferentes subredes

La misma situación ocurre cuando se utilizan interfaces de túnel virtual (VTI) (túnel de encapsulación de routing genérico (GRE)).

Topología: R1(Tun1: 172.16.0.1/24)—(Tun1: 172.17.0.2/24) R2

```
R1:
interface Ethernet0/0
 ip address 10.0.0.1 255.255.255.0

interface Tunnel1
 ip address 172.16.0.1 255.255.255.0
 tunnel source Ethernet0/0
 tunnel destination 10.0.0.2

router eigrp 100
 network 172.16.0.1 0.0.0.0
 passive-interface default
 no passive-interface Tunnel1

R2:
interface Ethernet0/0
 ip address 10.0.0.2 255.255.255.0

interface Tunnel1
 ip address 172.17.0.2 255.255.255.0
 tunnel source Ethernet0/0
 tunnel destination 10.0.0.1

router eigrp 100
 network 172.17.0.2 0.0.0.0
 passive-interface default
 no passive-interface Tunnel1
```

R1 muestra:

```
*Mar  3 16:41:52.167: EIGRP: Received HELLO on Tunnel1 nbr 172.17.0.2
*Mar  3 16:41:52.167:   AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0
*Mar  3 16:41:52.167: EIGRP-IPv4(100): Neighbor 172.17.0.2 not on common subnet
```

for Tunnel1
Debe ocurrir lo siguiente.

Utilice el comando IP Unnumbered

Este ejemplo muestra cómo utilizar el comando **ip unnumbered** que inhabilita la verificación y permite el establecimiento de una sesión EIGRP entre peers en diferentes subredes.

La topología es similar al ejemplo anterior, pero las direcciones de los túneles se definen ahora a través del comando **ip unnumbered** que señala a los loopbacks:

Topología: R1(Tun1: 172.16.0.1/24)—(Tun1: 172.17.0.2/24) R2

R1:
```
interface Ethernet0/0
 ip address 10.0.0.1 255.255.255.0

interface Loopback0
 ip address 172.16.0.1 255.255.255.0

interface Tunnel1
 ip unnumbered Loopback0
 tunnel source Ethernet0/0
 tunnel destination 10.0.0.2

router eigrp 100
 network 172.16.0.1 0.0.0.0
 passive-interface default
 no passive-interface Tunnel1
```

R2:
```
interface Ethernet0/0
 ip address 10.0.0.2 255.255.255.0

interface Loopback0
 ip address 172.17.0.2 255.255.255.0

interface Tunnel1
 ip unnumbered Loopback0
 tunnel source Ethernet0/0
 tunnel destination 10.0.0.1

router eigrp 100
 network 172.17.0.2 0.0.0.0
 passive-interface default
 no passive-interface Tunnel1
```

R1 muestra:

```
*Mar  3 16:50:39.046: EIGRP: Received HELLO on Tunnel1 nbr 172.17.0.2
*Mar  3 16:50:39.046:   AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0
*Mar  3 16:50:39.046: EIGRP: New peer 172.17.0.2
*Mar  3 16:50:39.046: %DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 172.17.0.2
(Tunnel1) is up: new adjacency

R1#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface       Hold Uptime   SRTT   RTO  Q  Seq
```

```
                                          (sec)         (ms)       Cnt Num
0   172.17.0.2              Tu1               12 00:00:07    7  1434  0  13

R1#show ip route eigrp
     172.17.0.0/24 is subnetted, 1 subnets
D        172.17.0.0 [90/27008000] via 172.17.0.2, 00:00:05, Tunnel1

R1#show ip int brief
Interface                  IP-Address      OK? Method Status                Protocol
Ethernet0/0                10.0.0.1         YES manual up                    up
Loopback0                172.16.0.1      YES manual up                    up
Tunnel1                    172.16.0.1      YES TFTP  up                     up
```

R2 es similar a esto.

Después de cambiar el comando **ip unnumbered** en una configuración de dirección IP específica, no se forma una adyacencia EIGRP.

EIGRP en SVTI para el segmento DVTI con diferentes subredes

Este ejemplo también utiliza el comando ip unnumbered. Las reglas mencionadas anteriormente también se aplican a DVTI.

Topología: R1(Tun1: 172.16.0.1/24)—(Virtual-template: 172.17.0.2/24) R2

El ejemplo anterior se modifica aquí para utilizar DVTI en lugar de SVTI. Además, en este ejemplo se agrega protección de túnel.

R1:

```
crypto isakmp policy 1
 encr 3des
 authentication pre-share
 group 2
crypto isakmp key cisco address 0.0.0.0 0.0.0.0
!
crypto ipsec transform-set TS esp-des esp-md5-hmac
!
crypto ipsec profile prof
 set transform-set TS
!
interface Loopback0
 ip address 172.16.0.1 255.255.255.0
!
interface Tunnel1
 ip unnumbered Loopback0
 tunnel source Ethernet0/0
 tunnel mode ipsec ipv4
 tunnel destination 10.0.0.2
 tunnel protection ipsec profile prof
!
router eigrp 100
 network 172.16.0.1 0.0.0.0
 passive-interface default
 no passive-interface Tunnel1
```

R2:

```
crypto isakmp policy 1
 encr 3des
```

```
 authentication pre-share
 group 2
crypto isakmp key cisco address 0.0.0.0 0.0.0.0
crypto isakmp profile profLAN
  keyring default
  match identity address 10.0.0.1 255.255.255.255
   virtual-template 1
!
crypto ipsec transform-set TS esp-des esp-md5-hmac
!
crypto ipsec profile profLAN
 set transform-set TS
 set isakmp-profile profLAN

interface Loopback0
 ip address 172.17.0.2 255.255.255.0
!
interface Ethernet0/0
 ip address 10.0.0.2 255.255.255.0
!
interface Virtual-Template1 type tunnel
 ip unnumbered Loopback0
 tunnel source Ethernet0/0
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile profLAN
!
!
router eigrp 100
 network 172.17.0.2 0.0.0.0
 passive-interface default
 no passive-interface Virtual-Template1
```

Todo funciona como se espera:

```
R1#show crypto session
Crypto session current status
Interface: Tunnel1
Session status: UP-ACTIVE
Peer: 10.0.0.2 port 500
 IKEv1 SA: local 10.0.0.1/500 remote 10.0.0.2/500 Active
 IPSEC FLOW: permit ip 0.0.0.0/0.0.0.0 0.0.0.0/0.0.0.0
       Active SAs: 2, origin: crypto map


R1#show crypto ipsec sa
interface: Tunnel1
   Crypto map tag: Tunnel1-head-0, local addr 10.0.0.1
  protected vrf: (none)
  local  ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
  remote ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
  current_peer 10.0.0.2 port 500
    PERMIT, flags={origin_is_acl,}
   #pkts encaps: 89, #pkts encrypt: 89, #pkts digest: 89
   #pkts decaps: 91, #pkts decrypt: 91, #pkts verify: 91


R1#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface       Hold Uptime   SRTT   RTO  Q  Seq
0   172.17.0.2              Tu1               13 00:06:31    7  1434  0  19

R1#show ip route eigrp
     172.17.0.0/24 is subnetted, 1 subnets
```

```
D        172.17.0.0 [90/27008000] via 172.17.0.2, 00:06:35, Tunnel1
```

```
R2#show crypto session
Crypto session current status
Interface: Virtual-Access1
Profile: profLAN
Session status: UP-ACTIVE
Peer: 10.0.0.1 port 500
 IKEv1 SA: local 10.0.0.2/500 remote 10.0.0.1/500 Active
 IPSEC FLOW: permit ip 0.0.0.0/0.0.0.0 0.0.0.0/0.0.0.0
       Active SAs: 2, origin: crypto map
```

```
R2#show crypto ipsec sa
interface: Virtual-Access1
   Crypto map tag: Virtual-Access1-head-0, local addr 10.0.0.2
  protected vrf: (none)
  local  ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
  remote ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
  current_peer 10.0.0.1 port 500
    PERMIT, flags={origin_is_acl,}
   #pkts encaps: 107, #pkts encrypt: 107, #pkts digest: 107
   #pkts decaps: 105, #pkts decrypt: 105, #pkts verify: 105
```

```
R2#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface       Hold Uptime   SRTT   RTO  Q  Seq
0   172.16.0.1              Vi1               13 00:07:41   11   200  0  16
```

```
R2#show ip route eigrp
     172.16.0.0/24 is subnetted, 1 subnets
D        172.16.0.0 [90/1433600] via 172.16.0.1, 00:07:44, Virtual-Access1
```

Como en los ejemplos anteriores, cuando intenta configurar 172.16.0.1 y 172.17.0.2 directamente bajo las interfaces de túnel, EIGRP falla con exactamente el mismo error que antes.

EIGRP en IKEv2 Flex VPN con diferentes subredes

Este es el ejemplo de la configuración de FlexVPN Hub y Spoke. El servidor envía la dirección IP a través del modo de configuración para el cliente.

Topología: R1(e0/0: 172.16.0.1/24)—(e0/1: 172.16.0.2/24) R2

Configuración del concentrador (R1):

```
aaa new-model
aaa authorization network LOCALIKEv2 local

crypto ikev2 authorization policy AUTHOR-POLICY
 pool POOL
!
```

```
crypto ikev2 keyring KEYRING
 peer R2
 address 172.16.0.2
 pre-shared-key CISCO
 !

crypto ikev2 profile default
 match identity remote key-id FLEX
 authentication remote pre-share
 authentication local pre-share
 keyring local KEYRING
 aaa authorization group psk list LOCALIKEv2 AUTHOR-POLICY
 virtual-template 1

interface Loopback0
 ip address 1.1.1.1 255.255.255.0
!
interface Ethernet0/0
 ip address 172.16.0.1 255.255.255.0

interface Virtual-Template1 type tunnel
 ip unnumbered Loopback0
 tunnel source Ethernet0/0
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile default
!
!
router eigrp 1
 network 1.1.1.1 0.0.0.0
 passive-interface default
 no passive-interface Virtual-Template1
!
ip local pool POOL 192.168.0.1 192.168.0.10
```

Configuración de radio:

```
aaa new-model
aaa authorization network FLEX local

crypto ikev2 authorization policy FLEX
 route set interface
!
!
!
crypto ikev2 keyring KEYRING
 peer R1
 address 172.16.0.1
 pre-shared-key CISCO
 !
!
!
crypto ikev2 profile default
 match identity remote address 172.16.0.1 255.255.255.255
 identity local key-id FLEX
 authentication remote pre-share
 authentication local pre-share
 keyring local KEYRING
 aaa authorization group psk list FLEX FLEX

interface Loopback0
 ip address 2.2.2.2 255.255.255.0
```

```
!
interface Ethernet0/0
 ip address 172.16.0.2 255.255.255.0

interface Tunnel0
 ip address negotiated
 tunnel source Ethernet0/0
 tunnel mode ipsec ipv4
 tunnel destination 172.16.0.1
 tunnel protection ipsec profile default

router eigrp 1
 network 0.0.0.0
 passive-interface default
 no passive-interface Tunnel0
```

El Spoke utiliza SVTI para conectarse al Hub que utiliza DVTI para todos los radios. Debido a que EIGRP no es tan flexible como Open Shortest Path First (OSPF) y no es posible configurarlo en la interfaz (SVTI o DVTI), la **red 0.0.0.0** se utiliza en Spoke para asegurarse de que EIGRP esté habilitado en la interfaz **Tun0**. Se utiliza una interfaz pasiva para asegurar que la adyacencia se forme solamente en la interfaz **Tun0**.

Para esta implementación, también es necesario configurar **ip unnumbered** en el Hub. Cuando configura manualmente una dirección IP bajo la interfaz de plantilla virtual, no se clona en la interfaz de acceso virtual. A continuación, la interfaz de acceso virtual no tiene asignada una dirección IP y la adyacencia EIGRP no se forma. Esta es la razón por la que el comando **ip unnumbered** siempre se requiere para las interfaces DVTI para formar una adyacencia EIGRP.

En este ejemplo, se genera una adyacencia EIGRP entre 1.1.1.1 y 192.168.0.9.

Prueba en el concentrador:

```
R1#show ip int brief
Interface              IP-Address      OK? Method Status                Protocol
Ethernet0/0            172.16.0.1      YES NVRAM  up                    up
Ethernet0/1            unassigned      YES NVRAM  administratively down down
Ethernet0/2            unassigned      YES NVRAM  administratively down down
Ethernet0/3            unassigned      YES NVRAM  administratively down down
Loopback0              1.1.1.1         YES manual up                    up
Virtual-Access1        1.1.1.1         YES unset  up                    up
Virtual-Template1      1.1.1.1         YES manual up                    down

R1#show crypto session
Crypto session current status

Interface: Virtual-Access1
Session status: UP-ACTIVE
Peer: 172.16.0.2 port 500
 IKEv2 SA: local 172.16.0.1/500 remote 172.16.0.2/500 Active
 IPSEC FLOW: permit ip 0.0.0.0/0.0.0.0 0.0.0.0/0.0.0.0
       Active SAs: 2, origin: crypto map

R1#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(1)
H   Address             Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                               (sec)         (ms)       Cnt Num
0   192.168.0.9          Vi1                      10 01:28:49   12  1494  0  13

R1#show ip route eigrp
....
```

```
Gateway of last resort is not set

     2.0.0.0/24 is subnetted, 1 subnets
D        2.2.2.0 [90/27008000] via 192.168.0.9, 01:28:52, Virtual-Access1
```

Desde la perspectiva Spoke, el comando **ip address negotiated** funciona igual que el comando **ip address unnumbered**, y la verificación de la subred está inhabilitada.

Probando en Spoke:

```
R2#show ip int brief
Interface              IP-Address      OK? Method Status                Protocol
Ethernet0/0            172.16.0.2      YES NVRAM  up                    up
Ethernet0/1            unassigned      YES NVRAM  administratively down down
Ethernet0/2            unassigned      YES NVRAM  administratively down down
Ethernet0/3            unassigned      YES NVRAM  administratively down down
Loopback0              2.2.2.2         YES NVRAM  up                    up
Tunnel0                192.168.0.9     YES NVRAM  up                    up

R2#show crypto session
Crypto session current status

Interface: Tunnel0
Session status: UP-ACTIVE
Peer: 172.16.0.1 port 500
 IKEv2 SA: local 172.16.0.2/500 remote 172.16.0.1/500 Active
 IPSEC FLOW: permit ip 0.0.0.0/0.0.0.0 0.0.0.0/0.0.0.0
       Active SAs: 2, origin: crypto map

R2#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(1)
H   Address             Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                               (sec)         (ms)       Cnt Num
0   1.1.1.1             Tu0                      14 01:30:18   15  1434  0  14

R2#show ip route eigrp
....
     1.0.0.0/24 is subnetted, 1 subnets
D        1.1.1.0 [90/27008000] via 1.1.1.1, 01:30:21
```

Modo de configuración para routing

Internet Key Exchange versión 2 (IKEv2) es otra opción. Es posible utilizar el modo de configuración para enviar rutas. En este escenario, no se necesitan EIGRP y el comando **ip unnumbered**.

Puede modificar el ejemplo anterior para configurar el Hub para que envíe esa ruta a través del modo de configuración:

```
crypto ikev2 authorization policy AUTHOR-POLICY
```

```
 pool POOL
 route set access-list SPLIT

ip access-list standard SPLIT
 permit 1.1.1.0 0.0.0.255
```

El Spoke ve 1.1.1.1 como estático, no como EIGRP:

```
R2#show ip route
....
     1.0.0.0/24 is subnetted, 1 subnets
S        1.1.1.0 is directly connected, Tunnel0
```

El mismo proceso funciona en la dirección opuesta. El radio envía una ruta al concentrador:

```
crypto ikev2 authorization policy FLEX
 route set access-list SPLIT

ip access-list standard SPLIT
 permit 2.2.2.0 0.0.0.255
```

El concentrador lo ve como estático (no EIGRP):

```
R1#show ip route
....
     2.0.0.0/24 is subnetted, 1 subnets
S        2.2.2.0 is directly connected, Virtual-Access1
```

Para este escenario, el protocolo de ruteo dinámico y el comando **ip unnumbered** no son necesarios.

EIGRP IPV6 en segmento SVTI con diferentes subredes

Para IPv6, la situación es diferente. Esto se debe a que las direcciones locales de link IPv6 (FE80::/10) se utilizan para generar la adyacencia EIGRP o OSPF. Las direcciones locales de link válidas siempre pertenecen a la misma subred, por lo que no hay necesidad de utilizar el comando **ipv6 no numerado** para eso.

La topología aquí es la misma que en el ejemplo anterior, excepto que todas las direcciones IPv4 se reemplazan por direcciones IPv6.

Configuración R1:

```
interface Tunnel1
 no ip address
 ipv6 address FE80:1::1 link-local
 ipv6 address 2001:1::1/64
 ipv6 enable
 ipv6 eigrp 100
 tunnel source Ethernet0/0
 tunnel mode gre ipv6
 tunnel destination 2001::2
```

```
interface Loopback0
 description Simulate LAN
 no ip address
 ipv6 address 2001:100::1/64
 ipv6 enable
 ipv6 eigrp 100

interface Ethernet0/0
 no ip address
 ipv6 address 2001::1/64
 ipv6 enable

ipv6 router eigrp 100
```

Configuración R2:

```
interface Tunnel1
 no ip address
 ipv6 address FE80:2::2 link-local
 ipv6 address 2001:2::2/64
 ipv6 enable
 ipv6 eigrp 100
 tunnel source Ethernet0/0
 tunnel mode gre ipv6
 tunnel destination 2001::1

interface Loopback0
 description Simulate LAN
 no ip address
 ipv6 address 2001:200::1/64
 ipv6 enable
 ipv6 eigrp 100

interface Ethernet0/0
 no ip address
 ipv6 address 2001::2/64
 ipv6 enable

ipv6 router eigrp 100
```

Las direcciones del túnel están en diferentes subredes (2001:1::1/64 y 2001:2::2/64), pero eso no es importante. Las direcciones locales de link se utilizan para generar adyacencia. Con estas direcciones, siempre tiene éxito.

En R1:

```
R1#show ipv6 int brief
Ethernet0/0            [up/up]
   FE80::A8BB:CCFF:FE00:6400
   2001::1
Loopback0              [up/up]
   FE80::A8BB:CCFF:FE00:6400
   2001:100::1
Tunnel1                [up/up]
   FE80:1::1
   2001:1::1

R1#show ipv6 eigrp neighbors
EIGRP-IPv6 Neighbors for AS(100)
H   Address             Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                (sec)         (ms)       Cnt Num
```

```
0   Link-local address: Tu1                     12 00:13:58  821  4926  0  17
    FE80:2::2

R1#show ipv6 route eigrp
...
D   2001:2::/64 [90/28160000]
    via FE80:2::2, Tunnel1
D   2001:200::/64 [90/27008000]
    via FE80:2::2, Tunnel1
```

En R2:

```
R2#show ipv6 int brief
Ethernet0/0            [up/up]
   FE80::A8BB:CCFF:FE00:6500
   2001::2
Loopback0              [up/up]
   FE80::A8BB:CCFF:FE00:6500
   2001:200::1
Tunnel1                [up/up]
   FE80:2::2
   2001:2::2

R2#show ipv6 eigrp neighbors
EIGRP-IPv6 Neighbors for AS(100)
H   Address             Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                               (sec)         (ms)       Cnt Num
0   Link-local address: Tu1                      14 00:15:31   21  1470  0  18
   FE80:1::1

R2#show ipv6 route eigrp
...
D   2001:1::/64 [90/28160000]
    via FE80:1::1, Tunnel1
D   2001:100::/64 [90/27008000]
    via FE80:1::1, Tunnel1
```

El proceso EIGRP instala la red IPv6 del par. En R1, la red 2001:2::/64 está instalada y esa red es una subred diferente a 2001:1::/64. Lo mismo es válido para R2. Por ejemplo, 2001::1/64 está instalado, que es una subred para su dirección IP de peer. No es necesario el comando **ipv6 unnumbered** aquí. Además, el comando **ipv6 address** no es necesario en la interfaz de túnel para establecer la adyacencia EIGRP, porque se utilizan las direcciones locales de link (y se generan automáticamente cuando se habilita IPv6 con el **comando ipv6 enable**).

EIGRP IPV6 en IKEv2 Flex VPN con diferentes subredes

La configuración DVTI para IPv6 es diferente a la de IPv4: ya no es posible configurar una dirección IP estática.

```
R1(config)#interface Virtual-Template2 type tunnel
R1(config-if)#ipv6 enable
R1(config-if)#ipv6 address ?
 autoconfig  Obtain address using autoconfiguration
 dhcp        Obtain a ipv6 address using dhcp
 negotiated  IPv6 Address negotiated via IKEv2 Modeconfig
```

```
R1(config-if)#ipv6 address
```

Esto se espera, ya que una dirección estática nunca se clona en una interfaz de acceso virtual. Por este motivo se recomienda el comando **ipv6 no numbered** para la configuración del hub, y el comando **ipv6 address negotiated** se recomienda para la configuración de Spoke.

La topología es la misma que en el ejemplo anterior, excepto que todas las direcciones IPv4 se reemplazan por direcciones IPv6.

Configuración del concentrador (R1):

```
aaa authorization network LOCALIKEv2 local

crypto ikev2 authorization policy AUTHOR-POLICY
 ipv6 pool POOL

crypto ikev2 keyring KEYRING
 peer R2
 address 2001::2/64
 pre-shared-key CISCO

crypto ikev2 profile default
 match identity remote key-id FLEX
 authentication remote pre-share
 authentication local pre-share
 keyring local KEYRING
 aaa authorization group psk list LOCALIKEv2 AUTHOR-POLICY
 virtual-template 1

interface Loopback0
 no ip address
 ipv6 address 2001:100::1/64
 ipv6 enable
 ipv6 eigrp 100

interface Ethernet0/0
 no ip address
 ipv6 address 2001::1/64
 ipv6 enable

interface Virtual-Template1 type tunnel
 no ip address
 ipv6 unnumbered Loopback0
 ipv6 enable
 ipv6 eigrp 100
 tunnel source Ethernet0/0
 tunnel mode ipsec ipv6
 tunnel protection ipsec profile default

ipv6 local pool POOL 2001:10::/64 64
ipv6 router eigrp 100
 eigrp router-id 1.1.1.1
```

Configuración de radio (R2):

```
aaa authorization network FLEX local

crypto ikev2 authorization policy FLEX
```

```
 route set interface

crypto ikev2 keyring KEYRING
 peer R1
 address 2001::1/64
 pre-shared-key CISCO

crypto ikev2 profile default
 match identity remote address 2001::1/64
 identity local key-id FLEX
 authentication remote pre-share
 authentication local pre-share
 keyring local KEYRING
 aaa authorization group psk list FLEX FLEX

interface Tunnel0
 no ip address
 ipv6 address negotiated
 ipv6 enable
 ipv6 eigrp 100
 tunnel source Ethernet0/0
 tunnel mode ipsec ipv6
 tunnel destination 2001::1
 tunnel protection ipsec profile default
!
interface Ethernet0/0
 no ip address
 ipv6 address 2001::2/64
 ipv6 enable

ipv6 router eigrp 100
 eigrp router-id 2.2.2.2
```

Verificación:

```
R2#show ipv6 eigrp neighbors
EIGRP-IPv6 Neighbors for AS(100)
H   Address             Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                              (sec)         (ms)       Cnt Num
0   Link-local address: Tu0                      11 00:12:32   17  1440  0  12
   FE80::A8BB:CCFF:FE00:6500

R2#show ipv6 route eigrp
....
D   2001:100::/64 [90/27008000]
    via FE80::A8BB:CCFF:FE00:6500, Tunnel0

R2#show crypto session detail
Crypto session current status

Code: C - IKE Configuration mode, D - Dead Peer Detection
K - Keepalives, N - NAT-traversal, T - cTCP encapsulation
X - IKE Extended Authentication, F - IKE Fragmentation

Interface: Tunnel0
Uptime: 00:13:17
Session status: UP-ACTIVE
Peer: 2001::1 port 500 fvrf: (none) ivrf: (none)
     Phase1_id: 2001::1
     Desc: (none)
 IKEv2 SA: local 2001::2/500
         remote 2001::1/500 Active
         Capabilities:(none) connid:1 lifetime:23:46:43
```

```
 IPSEC FLOW: permit ipv6 ::/0 ::/0
       Active SAs: 2, origin: crypto map
       Inbound:  #pkts dec'ed 190 drop 0 life (KB/Sec) 4271090/2803
       Outbound: #pkts enc'ed 194 drop 0 life (KB/Sec) 4271096/2803

R2#ping 2001:100::1 repeat 100
Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 2001:100::1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (100/100), round-trip min/avg/max = 1/4/5 ms

R2#show crypto session detail
Crypto session current status

Code: C - IKE Configuration mode, D - Dead Peer Detection
K - Keepalives, N - NAT-traversal, T - cTCP encapsulation
X - IKE Extended Authentication, F - IKE Fragmentation

Interface: Tunnel0
Uptime: 00:13:27
Session status: UP-ACTIVE
Peer: 2001::1 port 500 fvrf: (none) ivrf: (none)
     Phase1_id: 2001::1
     Desc: (none)
 IKEv2 SA: local 2001::2/500
         remote 2001::1/500 Active
         Capabilities:(none) connid:1 lifetime:23:46:33
 IPSEC FLOW: permit ipv6 ::/0 ::/0
       Active SAs: 2, origin: crypto map
       Inbound:  #pkts dec'ed 292 drop 0 life (KB/Sec) 4271071/2792
       Outbound: #pkts enc'ed 296 drop 0 life (KB/Sec) 4271082/2792
```

Para DVTI, IPv6 no se puede configurar manualmente. Se recomienda el comando **ipv6 no numbered** para el concentrador y el comando **ipv6 address negotiated** se recomienda en el radio.

Este escenario presenta el comando **ipv6 unnumbered** para DVTI. Es importante observar que, para IPv6 en lugar de IPv4, el comando **ipv6 no numerado** en la interfaz de plantilla virtual no es necesario. El motivo es el mismo que para el escenario SVTI de IPv6: la dirección ipv6 local de link se utiliza para generar adyacencia. La interfaz de acceso virtual, clonada a partir de la plantilla virtual, hereda la dirección local de link IPv6 y eso es suficiente para generar la adyacencia EIGRP.

Verificación

Actualmente, no hay un procedimiento de verificación disponible para esta configuración.

Troubleshoot

Actualmente, no hay información específica de troubleshooting disponible para esta configuración.

Advertencias conocidas

Id. de bug Cisco CSCtx45062 FlexVPN: Eigrp no debe verificar las subredes comunes si las ip del túnel son /32.

Este error y corrección no es específico de FlexVPN. Ingrese este comando antes de implementar la corrección (Software Release 15.1):

```
R2(config-if)#do show run int tun1
Building configuration...

Current configuration : 165 bytes

interface Tunnel1
 tunnel source Ethernet0/0
 tunnel destination 192.168.0.1
 tunnel protection ipsec profile prof1

R2(config-if)#ip address 192.168.200.1 255.255.255.255
Bad mask /32 for address 192.168.200.1
```

Ingrese este comando después de la corrección (software 15.3):

```
R2(config-if)#do show run int tun1
Building configuration...

Current configuration : 165 bytes

interface Tunnel1
 tunnel source Ethernet0/0
 tunnel destination 192.168.0.1
 tunnel protection ipsec profile prof1

R2(config-if)#ip address 192.168.200.1 255.255.255.255
R2(config-if)#
*Jun 14 18:01:12.395: %DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor
192.168.100.1 (Tunnel1) is up: new adjacency
```

En realidad hay dos cambios en la versión 15.3 del software:

- **Netmask /32** se acepta para todas las direcciones IP.
- No hay verificación de subred para un vecino EIGRP cuando utiliza la dirección **/32**.

Summary

El comportamiento de EIGRP se modifica por el comando **ip unnumbered**. Inhabilita las verificaciones de la misma subred mientras establece una adyacencia EIGRP.

También es importante recordar que cuando utiliza direcciones IP configuradas estáticamente en la plantilla virtual, no se clona en el acceso virtual. Esta es la razón por la que el comando **ip unnumbered** es necesario.

Para FlexVPN, no es necesario utilizar el comando **ip unnumbered** cuando se utiliza la dirección negociada en el cliente. Pero, es importante utilizarlo en el Hub cuando utilice EIGRP. Cuando utiliza el modo de configuración para el ruteo, no se necesita EIGRP.

Para SVTI, IPv6 utiliza direcciones locales de link para la adyacencia, y no hay necesidad de utilizar el comando **ipv6 unnumbered**.

Para DVTI, IPv6 no se puede configurar manualmente. Se recomienda el comando **ipv6 no numbered** para el concentrador y el comando **ipv6 address negotiated** se recomienda en el radio.

Información Relacionada

- Guía de configuración de FlexVPN de Cisco IOS 15.3
- Referencias de Comandos de Cisco IOS 15.3
- Soporte Técnico y Documentación - Cisco Systems