

# Exemplo de Configuração do EIGRP em SVTI, DVTI e IKEv2 FlexVPN com o Comando "IP[v6] Unnumbered"

## Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[EIGRP em um segmento Ethernet com sub-redes diferentes](#)

[EIGRP no segmento SVTI com sub-redes diferentes](#)

[Usar o comando IP Unnumbered](#)

[EIGRP no segmento SVTI para DVTI com sub-redes diferentes](#)

[EIGRP em VPN flexível IKEv2 com sub-redes diferentes](#)

[Modo de configuração para roteamento](#)

[IPV6 EIGRP no segmento SVTI com sub-redes diferentes](#)

[IPV6 EIGRP em VPN flexível IKEv2 com sub-redes diferentes](#)

[Verificar](#)

[Troubleshoot](#)

[Caveats conhecidos](#)

[Summary](#)

[Informações Relacionadas](#)

## Introduction

Este documento descreve como configurar o EIGRP (Enhanced Interior Gateway Routing Protocol) em vários cenários comuns no Cisco IOS®. Para aceitar uma adjacência de vizinhos EIGRP, o Cisco IOS deve obter o pacote HELLO do EIGRP de um endereço IP dentro da mesma sub-rede. É possível desativar essa verificação com o comando **ip unnumbered**.

A primeira parte do artigo apresenta uma falha do EIGRP quando ele recebe um pacote que não está na mesma sub-rede.

Outro exemplo demonstra o uso do comando **ip unnumbered** que desabilita essa verificação e permite que o EIGRP forme uma adjacência entre pares que pertencem a sub-redes diferentes.

Este artigo também apresenta uma implantação do FlexVPN Hub and Spoke com um endereço IP enviado do servidor. Para esse cenário, a verificação de sub-redes é desabilitada para o comando **ip address negotiation** e também para o comando **ip unnumbered**. O comando **ip unnumbered** é usado principalmente para interfaces do tipo Ponto a Ponto (P2P), e isso faz do FlexVPN uma

combinação perfeita, pois é baseado em uma arquitetura P2P.

Por fim, um cenário IPv6 é apresentado juntamente com as diferenças para Interfaces de Túnel Virtual Estático (SVTI - Static Virtual Tunnel Interfaces) e Interfaces de Túnel Virtual Dinâmico (DVTI - Dynamic Virtual Tunnel Interfaces). Há pequenas alterações no comportamento ao comparar cenários IPv6 com IPv4.

Além disso, as alterações entre as versões 15.1 e 15.3 do Cisco IOS são apresentadas ([ID de bug da Cisco CSCtx45062](#)).

O comando **ip unnumbered** é sempre necessário para DVTI. Isso ocorre porque os endereços IP configurados estaticamente em uma interface de modelo virtual nunca são clonados para uma interface de acesso virtual. Além disso, uma interface sem um endereço IP configurado não pode estabelecer nenhuma adjacência de protocolo de roteamento dinâmico. O comando **ip unnumbered** não é necessário para SVTI, mas sem essa sub-rede, a verificação é executada quando a adjacência do protocolo de roteamento dinâmico é estabelecida. Além disso, o comando **ipv6 unnumbered** não é necessário para cenários de IPV6 devido aos endereços link-local usados para criar adjacências de EIGRP.

## Prerequisites

### Requirements

A Cisco recomenda que você tenha conhecimento básico sobre estes tópicos:

- Configuração de VPN no Cisco IOS
- Configuração FlexVPN no Cisco IOS

### Componentes Utilizados

As informações neste documento são baseadas no Cisco IOS versão 15.3T.

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 em um segmento Ethernet com sub-redes diferentes

**Topologia:** Roteador 1 (R1) (e0/0: 10.0.0.1/24)—(e0/1: 10.0.1.2/24) Roteador 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 mostra:

```
*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
```

O Cisco IOS não forma uma adjacência, o que é esperado. Para obter mais informações sobre isso, consulte o [Significado das Mensagens de "Não em Sub-Rede Comum" do EIGRP?](#) artigo.

## EIGRP no segmento SVTI com sub-redes diferentes

A mesma situação ocorre quando você usa Virtual Tunnel Interfaces (VTI) (Generic Routing Encapsulation (GRE) Tunnel).

**Topologia:** R1(Tun1: 172.16.0.1/24)—(Toque1: 172.17.0.2/24) R2

**R1:**

```
interface Ethernet0/0
ip address 10.0.0.1 255.255.255.0

interface Tunnell
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 Tunnell
```

**R2:**

```
interface Ethernet0/0
ip address 10.0.0.2 255.255.255.0

interface Tunnell
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 Tunnell
```

R1 mostra:

```
*Mar 3 16:41:52.167: EIGRP: Received HELLO on Tunnell 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
```

Este é um comportamento esperado.

## Usar o comando IP Unnumbered

Este exemplo mostra como usar o comando **ip unnumbered** que desabilita a verificação e permite o estabelecimento de uma sessão EIGRP entre pares em sub-redes diferentes.

A topologia é semelhante ao exemplo anterior, mas os endereços dos túneis agora são definidos através do comando **ip unnumbered** que aponta para loopbacks:

**Topologia:** R1(Tun1: 172.16.0.1/24)—(Toque1: 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 mostra:

```
*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 (sec)	SRTT (ms)	RTO	Q	Seq 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 é semelhante a este.

Depois de alterar o comando **ip unnumbered** em uma configuração de endereço IP específica, uma adjacência EIGRP não se forma.

## EIGRP no segmento SVTI para DVTI com sub-redes diferentes

Este exemplo também usa o comando **ip unnumbered**. As regras mencionadas anteriormente aplicam-se também à DVTI.

**Topologia:** R1(Tun1: 172.16.0.1/24)—(Modelo virtual: 172.17.0.2/24) R2

O exemplo anterior é modificado aqui para usar DVTI em vez de SVTI. Além disso, a proteção de túnel é adicionada neste exemplo.

**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-Templatel 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-Templatel

```

Tudo funciona como esperado:

R1#**show crypto session**

```

Crypto session current status
Interface: Tunnell
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: Tunnell
  Crypto map tag: Tunnell-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
```

Quanto aos exemplos anteriores, quando você tenta configurar 172.16.0.1 e 172.17.0.2 diretamente nas interfaces de túnel, o EIGRP falha com exatamente o mesmo erro de antes.

## EIGRP em VPN flexível IKEv2 com sub-redes diferentes

Aqui está o exemplo da configuração do FlexVPN Hub and Spoke. O servidor envia o endereço IP através do modo de configuração para o cliente.

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

Configuração do hub (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-Templatel 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-Templatel
!
ip local pool POOL 192.168.0.1 192.168.0.10

```

## Configuração de spoke:

```

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

```

O Spoke usa SVTI para se conectar ao Hub que usa DVTI para todos os spokes. Como o EIGRP não é tão flexível quanto o OSPF (Open Shortest Path First) e não é possível configurá-lo na interface (SVTI ou DVTI), a **rede 0.0.0.0** é usada no Spoke para garantir que o EIGRP esteja ativado na interface **Tun0**. Uma interface passiva é usada para garantir que a adjacência seja formada somente na interface **Tun0**.

Para esta implantação, também é necessário configurar **ip não numerado** no Hub. Quando você configura manualmente um endereço IP na interface de modelo virtual, ele não é clonado na interface de acesso virtual. Em seguida, a interface de acesso virtual não tem um endereço IP atribuído e a adjacência EIGRP não se forma. É por isso que o comando **ip unnumbered** é sempre necessário para interfaces DVTI para formar uma adjacência EIGRP.

Neste exemplo, uma adjacência EIGRP é construída entre 1.1.1.1 e 192.168.0.9.

Teste no hub:

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	<b>1.1.1.1</b>	YES	unset	up	up
Virtual-Templatel	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 (sec)	Uptime	SRTT (ms)	RTO	Q Cnt	Seq Num
0	192.168.0.9	Vil	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

```

Da perspectiva de Spoke, o comando **ip address negotiation** funciona da mesma forma que o comando **ip address unnumbered**, e a verificação da sub-rede é desativada.

Testando no 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    administrativ down down
Ethernet0/2              unassigned     YES NVRAM    administrativ down down
Ethernet0/3              unassigned     YES NVRAM    administrativ 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 configuração para roteamento

O Internet Key Exchange versão 2 (IKEv2) é outra opção. É possível usar o modo de configuração para enviar rotas. Neste cenário, o EIGRP e o comando **ip unnumbered** não são necessários.

Você pode modificar o exemplo anterior para configurar o Hub para enviar essa rota através do modo de configuração:

```
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
```

O Spoke vê 1.1.1.1 como estático, não como EIGRP:

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

O mesmo processo funciona na direção oposta. O Spoke envia uma rota para o Hub:

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

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

O Hub o vê como estático (não 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 esse cenário, o protocolo de roteamento dinâmico e o comando **ip unnumbered** não são necessários.

## IPV6 EIGRP no segmento SVTI com sub-redes diferentes

Para IPv6, a situação é diferente. Isso ocorre porque os endereços de link local IPv6 (FE80::/10) são usados para criar adjacência de EIGRP ou OSPF. Os endereços link local válidos sempre pertencem à mesma sub-rede, portanto não há necessidade de usar o comando **ipv6 unnumbered** para isso.

A topologia aqui é a mesma do exemplo anterior, exceto que todos os endereços IPv4 são substituídos por endereços IPv6.

Configuração de 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
```

## Configuração do R2:

```
interface Tunnell
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
```

Os endereços de túnel estão em sub-redes diferentes (2001:1::1/64 e 2001:2::2/64), mas isso não é importante. Os endereços de link local são usados para criar adjacência. Com esses endereços, ele sempre é bem-sucedido.

## Em 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
Tunnell              [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
   FE80:2::2

```

```
R1#show ipv6 route eigrp
```

```

...
D 2001:2::/64 [90/28160000]
  via FE80:2::2, Tunnell
D 2001:200::/64 [90/27008000]
  via FE80:2::2, Tunnell

```

Em 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
Tunnell              [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
   FE80:1::1

```

```
R2#show ipv6 route eigrp
```

```

...
D 2001:1::/64 [90/28160000]
  via FE80:1::1, Tunnell
D 2001:100::/64 [90/27008000]
  via FE80:1::1, Tunnell

```

A rede IPv6 do peer é instalada pelo processo EIGRP. Em R1, a rede 2001:2::/64 está instalada e essa rede é uma sub-rede diferente de 2001:1::/64. O mesmo vale para R2. Por exemplo, 2001::1/64 está instalado, que é uma sub-rede para seu endereço IP de peer. Não há necessidade do comando **ipv6 unnumbered** aqui. Além disso, o comando **ipv6 address** não é necessário na interface do túnel para estabelecer a adjacência do EIGRP, pois os endereços link local são usados (e eles são gerados automaticamente quando você habilita o IPv6 com o comando **ipv6 enable**).

## IPv6 EIGRP em VPN flexível IKEv2 com sub-redes diferentes

A configuração DVTI para IPv6 é diferente da configuração para IPv4: não é mais possível configurar um endereço IP estático.

```

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
```

Isso é esperado, já que um endereço estático nunca é clonado para uma interface de acesso virtual. É por isso que o comando **ipv6 unnumbered** é recomendado para configuração de Hub, e o comando **ipv6 address negotiation** é recomendado para configuração de Spoke.

A topologia é a mesma do exemplo anterior, exceto que todos os endereços IPv4 são substituídos por endereços IPv6.

### Configuração do hub (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-Templatel 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
```

### Configuração de spoke (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

```

### Verificação:

#### R2#show ipv6 eigrp neighbors

```

EIGRP-IPv6 Neighbors for AS(100)

```

H	Address	Interface	Hold Uptime (sec)	SRTT (ms)	RTO	Q	Seq Cnt Num
0	Link-local address: Tu0 FE80::A8BB:CCFF:FE00:6500		11 00:12:32	17	1440	0	12

#### 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, o IPv6 não pode ser configurado manualmente. O comando **ipv6 unnumbered** é recomendado para o Hub, e o comando **ipv6 address negociado** é recomendado no Spoke.

Este cenário apresenta o comando **ipv6 unnumbered** para DVTI. É importante observar que, para IPv6 em vez de IPv4, o comando **ipv6 unnumbered** na interface de modelo virtual não é necessário. O motivo é o mesmo do cenário IPv6 SVTI: o endereço **ipv6 link local** é usado para criar adjacência. A interface de acesso virtual, que é clonada a partir do modelo virtual, herda o endereço link local IPv6 e isso é suficiente para criar adjacência EIGRP.

## Verificar

No momento, não há procedimento de verificação disponível para esta configuração.

## Troubleshoot

Atualmente, não existem informações disponíveis específicas sobre Troubleshooting para esta configuração.



# Caveats conhecidos

[ID de bug Cisco CSCtx45062](#) FlexVPN: O Eigrp não deve verificar sub-redes comuns se os ips de túnel forem /32.

Este bug e correção não são específicos para FlexVPN. Insira este comando antes de implementar a correção (Software Release 15.1):

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

Current configuration : 165 bytes

interface Tunnell
 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
```

Insira este comando após a correção (software 15.3):

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

Current configuration : 165 bytes

interface Tunnell
 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 (Tunnell) is up: new adjacency
```

Na verdade, há duas mudanças no Software Release 15.3:

- A máscara de rede /32 é aceita para todos os endereços IP.
- Não há verificação de sub-rede para um vizinho EIGRP quando você usa o endereço /32.

## Summary

O comportamento do EIGRP é alterado pelo comando **ip unnumbered**. Desativa verificações para a mesma sub-rede enquanto estabelece uma adjacência EIGRP.

Também é importante lembrar que, quando você usa DVTIs configurados estaticamente no modelo virtual, ele não é clonado para o acesso virtual. É por isso que o comando **ip unnumbered** é necessário.

Para o FlexVPN, não há necessidade de usar o comando **ip unnumbered** quando você usa o endereço negociado no cliente. Mas, é importante usá-lo no Hub quando você usa o EIGRP. Quando você usa o modo de configuração para roteamento, o EIGRP não é necessário.

Para SVTI, o IPv6 usa endereços de link local para adjacência e não há necessidade de usar o comando **ipv6 unnumbered**.

Para DVTI, o IPv6 não pode ser configurado manualmente. O comando **ipv6 unnumbered** é recomendado para o Hub, e o comando **ipv6 address negociado** é recomendado no Spoke.

## Informações Relacionadas

- [Guia de configuração do Cisco IOS 15.3 FlexVPN](#)
- [Referências de comando do Cisco IOS 15.3](#)
- [Suporte Técnico e Documentação - Cisco Systems](#)