

EIGRP auf SVTI, DVTI und IKEv2 FlexVPN mit dem Konfigurationsbeispiel "IP[v6] Unnumbered"-Befehl

Inhalt

[Einführung](#)

[Voraussetzungen](#)

[Anforderungen](#)

[Verwendete Komponenten](#)

[EIGRP auf einem Ethernet-Segment mit unterschiedlichen Subnetzen](#)

[EIGRP auf SVTI-Segment mit unterschiedlichen Subnetzen](#)

[Verwenden des Befehls IP Unnumbered \(nicht nummerierte IP\)](#)

[EIGRP auf SVTI zum DVTI-Segment mit unterschiedlichen Subnetzen](#)

[EIGRP auf IKEv2 Flex VPN mit unterschiedlichen Subnetzen](#)

[Konfigurationsmodus für Routing](#)

[IPV6 EIGRP auf SVTI-Segment mit unterschiedlichen Subnetzen](#)

[IPV6 EIGRP auf IKEv2 Flex VPN mit unterschiedlichen Subnetzen](#)

[Überprüfen](#)

[Fehlerbehebung](#)

[Bekannte Einwände](#)

[Zusammenfassung](#)

[Weitere Informationen](#)

Einführung

In diesem Dokument wird beschrieben, wie das Enhanced Interior Gateway Routing Protocol (EIGRP) in einer Reihe häufig auftretender Szenarien auf Cisco IOS[®] konfiguriert wird. Um eine EIGRP-Nachbarschaft zu akzeptieren, muss Cisco IOS das EIGRP-HELLO-Paket von einer IP-Adresse im gleichen Subnetz abrufen. Sie können diese Überprüfung mit dem Befehl **ip unnumbered (nicht nummerierte IP-Adresse)** deaktivieren.

Im ersten Teil des Artikels wird ein EIGRP-Fehler angezeigt, wenn ein Paket empfangen wird, das sich nicht im gleichen Subnetz befindet.

Ein weiteres Beispiel veranschaulicht die Verwendung des Befehls **ip unnumbered**, der diese Überprüfung deaktiviert und es EIGRP ermöglicht, eine Adjazenz zwischen Peers zu bilden, die verschiedenen Subnetzen angehören.

In diesem Artikel wird auch eine FlexVPN-Hub-and-Spoke-Bereitstellung mit einer vom Server gesendeten IP-Adresse vorgestellt. In diesem Szenario ist die Überprüfung von Subnetzen für den

Befehl **ip address negotiation** und auch für den Befehl **ip unnumbered** (**nicht nummerierte IP-Adresse**) deaktiviert. Der Befehl **ip unnumbered** wird hauptsächlich für Point-to-Point (P2P)-Typschnittstellen verwendet. Dies macht FlexVPN zu einem idealen Ort, da es auf einer P2P-Architektur basiert.

Schließlich wird ein IPv6-Szenario mit Unterschieden sowohl für die SVTI (Static Virtual Tunnel Interfaces) als auch für die DVTI (Dynamic Virtual Tunnel Interfaces) vorgestellt. Beim Vergleich von IPv6- mit IPv4-Szenarien treten geringfügige Verhaltensänderungen auf.

Außerdem werden die Änderungen zwischen den Cisco IOS Versionen 15.1 und 15.3 dargestellt ([Cisco Bug ID CSCtx45062](#)).

Der Befehl **ip unnumbered** (**nicht nummerierte IP**) wird für DVTI immer benötigt. Der Grund hierfür ist, dass statisch konfigurierte IP-Adressen auf einer Virtual-Template-Schnittstelle nie auf eine Virtual-Access-Schnittstelle geklont werden. Darüber hinaus kann eine Schnittstelle ohne konfigurierte IP-Adresse keine Adjacency für dynamisches Routing-Protokoll herstellen. Der Befehl **ip unnumbered** ist für SVTI nicht erforderlich. Ohne dieses Subnetz wird jedoch eine Überprüfung durchgeführt, wenn eine Adjacency für dynamische Routing-Protokolle eingerichtet wird. Der Befehl **ipv6 unnumbered** (**nicht nummerierte IPv6**) wird für IPV6-Szenarien wegen der lokalen Link-Adressen nicht benötigt, die zum Erstellen von EIGRP-Adjacencies verwendet werden.

Voraussetzungen

Anforderungen

Cisco empfiehlt, über grundlegende Kenntnisse in folgenden Bereichen zu verfügen:

- VPN-Konfiguration in Cisco IOS
- FlexVPN-Konfiguration auf Cisco IOS

Verwendete Komponenten

Die Informationen in diesem Dokument basieren auf Cisco IOS Version 15.3T.

Die Informationen in diesem Dokument wurden von den Geräten in einer bestimmten Laborumgebung erstellt. Alle in diesem Dokument verwendeten Geräte haben mit einer leeren (Standard-)Konfiguration begonnen. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die potenziellen Auswirkungen eines Befehls verstehen.

EIGRP auf einem Ethernet-Segment mit unterschiedlichen Subnetzen

Topologie: 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 zeigt:

```
*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 bildet keine Adjacency, wie erwartet. Weitere Informationen hierzu finden Sie im [Mean What Do EIGRP "Not On Common Subnet" \(Was soll EIGRP "Not On Common Subnet" \(Nicht über gemeinsames Subnetz\)\)?](#) Artikel.

EIGRP auf SVTI-Segment mit unterschiedlichen Subnetzen

Die gleiche Situation tritt auf, wenn Sie Virtual Tunnel Interfaces (VTI) (Generic Routing Encapsulation (GRE) Tunnel) verwenden.

Topologie: 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 zeigt:

```
*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
```

Dieses Verhalten wird erwartet.

Verwenden des Befehls IP Unnumbered (nicht nummerierte IP)

In diesem Beispiel wird veranschaulicht, wie der Befehl **ip unnumbered (unnummerierte)** verwendet wird, der die Überprüfung deaktiviert und die Einrichtung einer EIGRP-Sitzung zwischen Peers in verschiedenen Subnetzen ermöglicht.

Die Topologie ähnelt dem vorherigen Beispiel, aber die Adressen der Tunnel werden jetzt über den Befehl **ip unnumbered (unnummerierte IP-Adresse)** definiert, der auf Loopbacks zeigt:

Topologie: 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 zeigt:

```
*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 ähnelt dem hier.

Nachdem Sie den Befehl **ip unnumbered (nicht nummerierte IP)** in eine bestimmte IP-Adresskonfiguration geändert haben, bildet sich keine EIGRP-Adjacency aus.

EIGRP auf SVTI zum DVTI-Segment mit unterschiedlichen Subnetzen

In diesem Beispiel wird auch der Befehl **ip unnumbered (nicht nummerierte IP)** verwendet. Die oben genannten Regeln gelten auch für DVTI.

Topologie: R1 (Tun1): 172.16.0.1/24—(Virtuelle Vorlage: 172.17.0.2/24) R2

Das vorherige Beispiel wird hier geändert, um DVTI anstelle von SVTI zu verwenden. Zusätzlich wird in diesem Beispiel der Tunnelschutz hinzugefügt.

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 Tunnell

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

```

Alles funktioniert wie erwartet:

```

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

Wie bei den vorherigen Beispielen schlägt EIGRP bei der direkten Konfiguration von 172.16.0.1 und 172.17.0.2 unter den Tunnelschnittstellen mit dem gleichen Fehler wie zuvor fehl.

EIGRP auf IKEv2 Flex VPN mit unterschiedlichen Subnetzen

Hier ist das Beispiel für die FlexVPN-Hub-and-Spoke-Konfiguration. Der Server sendet die IP-Adresse über den Konfigurationsmodus für den Client.

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

Hub-Konfiguration (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
```

Spoke-Konfiguration:

```
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

```

Der Spoke verwendet SVTI, um eine Verbindung zum Hub herzustellen, der DVTI für alle Spokes verwendet. Da EIGRP nicht so flexibel wie Open Shortest Path First (OSPF) ist und nicht unter der Schnittstelle (SVTI oder DVTI) konfiguriert werden kann, wird das **Netzwerk 0.0.0.0** auf dem Spoke verwendet, um sicherzustellen, dass EIGRP auf der **Tun0**-Schnittstelle aktiviert ist. Eine passive Schnittstelle wird verwendet, um sicherzustellen, dass die Adjacency nur auf der **Tun0**-Schnittstelle gebildet wird.

Für diese Bereitstellung ist es auch erforderlich, **ip unnumbered** auf dem Hub zu konfigurieren. Wenn Sie eine IP-Adresse manuell unter der Virtual-Template-Schnittstelle konfigurieren, wird sie nicht auf die Virtual-Access-Schnittstelle geklont. Dann wird der virtuellen Zugriffsebene keine IP-Adresse zugewiesen, und die EIGRP-Adjacency bildet sich nicht. Aus diesem Grund ist der Befehl **ip unnumbered (nicht nummerierte IP)** immer für DVTI-Schnittstellen erforderlich, um eine EIGRP-Adjacency zu bilden.

In diesem Beispiel wird eine EIGRP-Adjacency zwischen 1.1.1.1 und 192.168.0.9 erstellt.

Tests am 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	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
```

Aus Spoke-Perspektive funktioniert der Befehl **ip address negotiation** genauso wie der Befehl **ip address unnumbered**, und die Überprüfung des Subnetzes ist deaktiviert.

Tests am 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
```

Konfigurationsmodus für Routing

Eine weitere Option ist Internet Key Exchange Version 2 (IKEv2). Es ist möglich, den Konfigurationsmodus zu verwenden, um Routen zu übertragen. In diesem Szenario sind EIGRP und der Befehl **ip unnumbered** nicht erforderlich.

Sie können das vorherige Beispiel ändern, um den Hub so zu konfigurieren, dass er diese Route im Konfigurationsmodus sendet:

```
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
```

Der Spoke sieht 1.1.1.1 als statisch, nicht als EIGRP:

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

Derselbe Prozess funktioniert in die entgegengesetzte Richtung. Der Spoke sendet eine Route zum 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
```

Der Hub sieht es als statisch (nicht EIGRP):

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

In diesem Szenario sind das dynamische Routing-Protokoll und der Befehl **ip unnumbered** (nicht nummerierte IP) nicht erforderlich.

IPv6 EIGRP auf SVTI-Segment mit unterschiedlichen Subnetzen

Bei IPv6 sieht die Situation anders aus. Dies liegt daran, dass IPv6-Link-Local-Adressen (FE80::/10) zum Erstellen einer EIGRP- oder OSPF-Adjacency verwendet werden. Gültige lokale Link-Adressen gehören immer zum gleichen Subnetz, daher ist es nicht erforderlich, den Befehl **ipv6 unnumbered** zu verwenden.

Die Topologie hier ist die gleiche wie im vorherigen Beispiel, mit der Ausnahme, dass alle IPv4-Adressen durch IPv6-Adressen ersetzt werden.

R1-Konfiguration:

```
interface Tunnell
  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
```

R2-Konfiguration:

```
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
```

Die Tunneladressen sind in verschiedenen Subnetzen (2001:1:1/64 und 2001:2:2/64), aber das ist nicht wichtig. Link-Local-Adressen werden verwendet, um Adjacency zu erstellen. Mit diesen Adressen ist es immer erfolgreich.

Auf 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
Tunnell1                 [up/up]
FE80:1::1
2001:1::1

```

R1#show ipv6 eigrp neighbors

EIGRP-IPv6 Neighbors for AS(100)

H	Address	Interface	Hold (sec)	Uptime	SRTT (ms)	RTO	Q Cnt	Seq Num
0	Link-local address: Tu1 FE80:2::2		12	00:13:58	821	4926	0	17

R1#show ipv6 route eigrp

```

...
D 2001:2::/64 [90/28160000]
  via FE80:2::2, Tunnell1
D 2001:200::/64 [90/27008000]
  via FE80:2::2, Tunnell1

```

Zu 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
Tunnell1                 [up/up]
FE80:2::2
2001:2::2

```

R2#show ipv6 eigrp neighbors

EIGRP-IPv6 Neighbors for AS(100)

H	Address	Interface	Hold (sec)	Uptime	SRTT (ms)	RTO	Q Cnt	Seq Num
0	Link-local address: Tu1 FE80:1::1		14	00:15:31	21	1470	0	18

R2#show ipv6 route eigrp

```

...
D 2001:1::/64 [90/28160000]
  via FE80:1::1, Tunnell1
D 2001:100::/64 [90/27008000]
  via FE80:1::1, Tunnell1

```

Das Peer-IPv6-Netzwerk wird vom EIGRP-Prozess installiert. Auf R1 ist das 2001:2:/64-Netzwerk installiert, das ein anderes Subnetz ist als 2001:1:/64. Gleiches gilt für R2. 2001::1/64 wird installiert, d. h. ein Subnetz für seine Peer-IP-Adresse. Hier ist der Befehl **ipv6 unnumbered (nicht nummerierte IPv6)** nicht erforderlich. Darüber hinaus wird der Befehl **ipv6 address** nicht für die Tunnelschnittstelle benötigt, um die EIGRP-Adjacency einzurichten, da link-local-Adressen verwendet werden (und diese werden automatisch generiert, wenn Sie IPv6 mit dem Befehl **ipv6 enable** aktivieren).

IPV6 EIGRP auf IKEv2 Flex VPN mit unterschiedlichen Subnetzen

Die DVTI-Konfiguration für IPv6 unterscheidet sich von der für IPv4: Es ist nicht mehr möglich, eine statische IP-Adresse zu konfigurieren.

```
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
```

Dies ist zu erwarten, da eine statische Adresse niemals auf eine Virtual-Access-Schnittstelle geklont wird. Aus diesem Grund wird der Befehl **ipv6 unnumbered** (nicht nummerierte IPv6) für die Hub-Konfiguration empfohlen, und der Befehl **ipv6 address negotiation** (ausgehandelte IPv6-Adresse) wird für die Spoke-Konfiguration empfohlen.

Die Topologie ist die gleiche wie im vorherigen Beispiel, mit der Ausnahme, dass alle IPv4-Adressen durch IPv6-Adressen ersetzt werden.

Hub-Konfiguration (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-Templat1 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
```

Spoke-Konfiguration (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
```

Überprüfung:

```
R2#show ipv6 eigrp neighbors
```

```
EIGRP-IPv6 Neighbors for AS(100)
```

H	Address	Interface	Hold (sec)	Uptime	SRTT (ms)	RTO	Q Cnt	Seq 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

Für DVTI kann IPv6 nicht manuell konfiguriert werden. Der Befehl **ipv6 unnumbered (nicht nummerierte IPv6)** wird für den Hub empfohlen, und der Befehl **ipv6 address negotiation (IPv6-Adresse wird ausgehandelt)** wird für den Spoke empfohlen.

Dieses Szenario stellt den **unnummerierten** Befehl **ipv6** für DVTI dar. Dabei ist zu beachten, dass für IPv6 und nicht für IPv4 der **unnummerierte** Befehl **ipv6** auf der Virtual-Template-Schnittstelle nicht benötigt wird. Der Grund dafür ist der gleiche wie für das IPv6 SVTI-Szenario: Die link-local ipv6-Adresse wird für die Erstellung der Adjacency verwendet. Die von der virtuellen Vorlage geklonte Virtual-Access-Schnittstelle erbt die IPv6-Link-Local-Adresse. Dies reicht aus, um eine

EIGRP-Adjacency zu erstellen.

Überprüfen

Für diese Konfiguration ist derzeit kein Überprüfungsverfahren verfügbar.

Fehlerbehebung

Für diese Konfiguration sind derzeit keine spezifischen Informationen zur Fehlerbehebung verfügbar.

Bekannte Einwände

[Cisco Bug-ID CSCtx45062](#) FlexVPN: EIGRP sollte keine allgemeinen Subnetze überprüfen, wenn tunnel ip's /32 sind.

Dieser Fehler und diese Behebung ist nicht FlexVPN-spezifisch. Geben Sie diesen Befehl ein, bevor Sie das Fix (Softwareversion 15.1) implementieren:

```
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
```

Geben Sie nach der Behebung (Software 15.3) den folgenden Befehl ein:

```
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
```

In Softwareversion 15.3 sind zwei Änderungen enthalten:

- **Die Netzmaske /32** wird für alle IP-Adressen akzeptiert.
- Wenn Sie die Adresse **/32** verwenden, wird für einen EIGRP-Nachbarn keine Subnetzüberprüfung durchgeführt.

Zusammenfassung

Das EIGRP-Verhalten wird durch den Befehl **ip unnumbered (nicht nummerierte IP)** geändert. Es deaktiviert Prüfungen für dasselbe Subnetz, während eine EIGRP-Adjacency erstellt wird.

Beachten Sie auch, dass bei Verwendung statisch konfigurierter IP-Adressen von DVTIs auf der virtuellen Vorlage kein Cloning auf den virtuellen Zugriff erfolgt. Aus diesem Grund wird der Befehl **ip unnumbered (nicht nummerierte IP)** benötigt.

Bei FlexVPN muss der Befehl **ip unnumbered (nicht nummerierte IP)** nicht verwendet werden, wenn die ausgehandelte Adresse auf dem Client verwendet wird. Es ist jedoch wichtig, dass Sie es auf dem Hub verwenden, wenn Sie EIGRP verwenden. Wenn Sie den Konfigurationsmodus für das Routing verwenden, ist EIGRP nicht erforderlich.

Für SVTIs verwendet IPv6 für die Adjacency link-local-Adressen, und der Befehl **ipv6 unnumbered (nicht nummerierte IPv6)** ist nicht erforderlich.

Für DVTI kann IPv6 nicht manuell konfiguriert werden. Der Befehl **ipv6 unnumbered (nicht nummerierte IPv6)** wird für den Hub empfohlen, und der Befehl **ipv6 address negotiation (IPv6-Adresse wird ausgehandelt)** wird für den Spoke empfohlen.

Weitere Informationen

- [Cisco IOS 15.3 FlexVPN-Konfigurationsleitfaden](#)
- [Cisco IOS 15.3-Befehlsreferenzen](#)
- [Technischer Support und Dokumentation - Cisco Systems](#)