

Fusierouter in SDA configureren

Inhoud

[Inleiding](#)

[Voorwaarden](#)

[Vereisten](#)

[Gebruikte componenten](#)

[Achtergrondinformatie](#)

[Functionaliteit van een fusieapparaat in DNA SD-Access-oplossing](#)

[Configureren](#)

[Netwerkdigram](#)

[Configuraties](#)

[Stap 1. De koppeling Hand-off vanuit DNAC configureren](#)

[Stap 2. Controleer de configuraties die door DNAC op border-routers worden uitgevoerd](#)

[Stap 3. Inloggen op border-routers configureren](#)

[Stap 4. Fusion-routers configureren](#)

[Stap 5. VRF-lekkage op fusierouterie configureren](#)

[Verifiëren](#)

[Stap 1. Controleer de eBGP-peer tussen fusie- en grensrouters](#)

[Stap 2. Controleer de iBGP-peer tussen beide Fusion-routers](#)

[Stap 3. Controleer prefixes in BGP-tabel en routingtabel](#)

[Handmatige configuratie voor border-redundantie](#)

[SDA-Border-1](#)

[SDA-Border-2](#)

[Vereenvoudig fusieconfiguratie met het gebruik van sjablonen](#)

[Definitie van variabelen](#)

[Voorbeeld van sjabloon](#)

[Fusie 1](#)

[Fusie 2](#)

Inleiding

In dit document wordt beschreven hoe u fusierouters configureert in een Cisco SDA-oplossing (Cisco Software-Defined Access of SD-Access).

Voorwaarden

Vereisten

Er zijn geen specifieke vereisten van toepassing op dit document.

Opmerking: installatie is vereist volgens Ondersteunde apparaten die u kunt vinden onder [Link to Releaseopmerkingen](#)

Gebruikte componenten

De informatie in dit document is gebaseerd op de volgende hardware-versies:

- DNAC, versie 1.2.1
- Rand en border - Cat3k Cisco-switch
- Fusie - Cisco-router met ondersteuning voor doorgifte tussen VRF's

De informatie in dit document is gebaseerd op de apparaten in een specifieke laboratoriumomgeving. Alle apparaten die in dit document worden beschreven, hadden een opgeschoonde (standaard)configuratie. Als uw netwerk live is, moet u zorgen dat u de potentiële impact van elke opdracht begrijpt.

Achtergrondinformatie

In de Cisco SD-Access-oplossing worden apparaten beheerd en geconfigureerd door Cisco DNA Center. Over het algemeen kunnen alle onderdelen van de SD-Access-fabric worden geconfigureerd en beheerd door Cisco DNA Center. Het fusieapparaat bevindt zich echter buiten de fabric en wordt dus handmatig geconfigureerd. Border Automation, hierna besproken, is een functie binnen Cisco DNA Center die de Border configuratie voor de overdracht van VRF's naar de Fusion-apparaten kan automatiseren.

Om redenen die gewoonlijk verband houden met de compatibiliteit met de huidige configuratie, is grensautomatisering soms niet geschikt, en dus kan de overdracht van de grens naar het fusieapparaat ook met de hand worden geconfigureerd. Inzicht in de gebruikte configuratie helpt belangrijke details te illustreren over de optimale configuratie en werking van het hele systeem.

Functionaliteit van een fusieapparaat in DNA SD-Access-oplossing

Met een fusieapparaat wordt doorgifte via VRF (Virtual Routing and Forwarding) in SD-Access-fabricdomeinen en hostconnectiviteit met gedeelde services, zoals DHCP, DNS, NTP, ISE, Cisco DNA Center en wireless LAN-controllers (WLC), mogelijk gemaakt. Hoewel deze rol door andere apparaten dan routers kan worden vervuld, wordt in dit document alleen gekeken naar routers als fusieapparaten.

Zoals eerder vermeld, moeten de gedeelde diensten ter beschikking worden gesteld van alle virtuele netwerken (VN) op de Campus. Dit wordt bereikt door de aanmaak van BGP-peerings (Border Gateway Protocol) van de grensrouters naar de Fusion Routers. Op de fusierouter worden de subnetten van de fabric-VRF die toegang moeten hebben tot deze services doorgegeven aan de GRT, of een VRF voor gedeelde services, en andersom. Routekaarten kunnen worden gebruikt om specifieke routingtabellen voor SD-Access-fabrics in op te nemen.

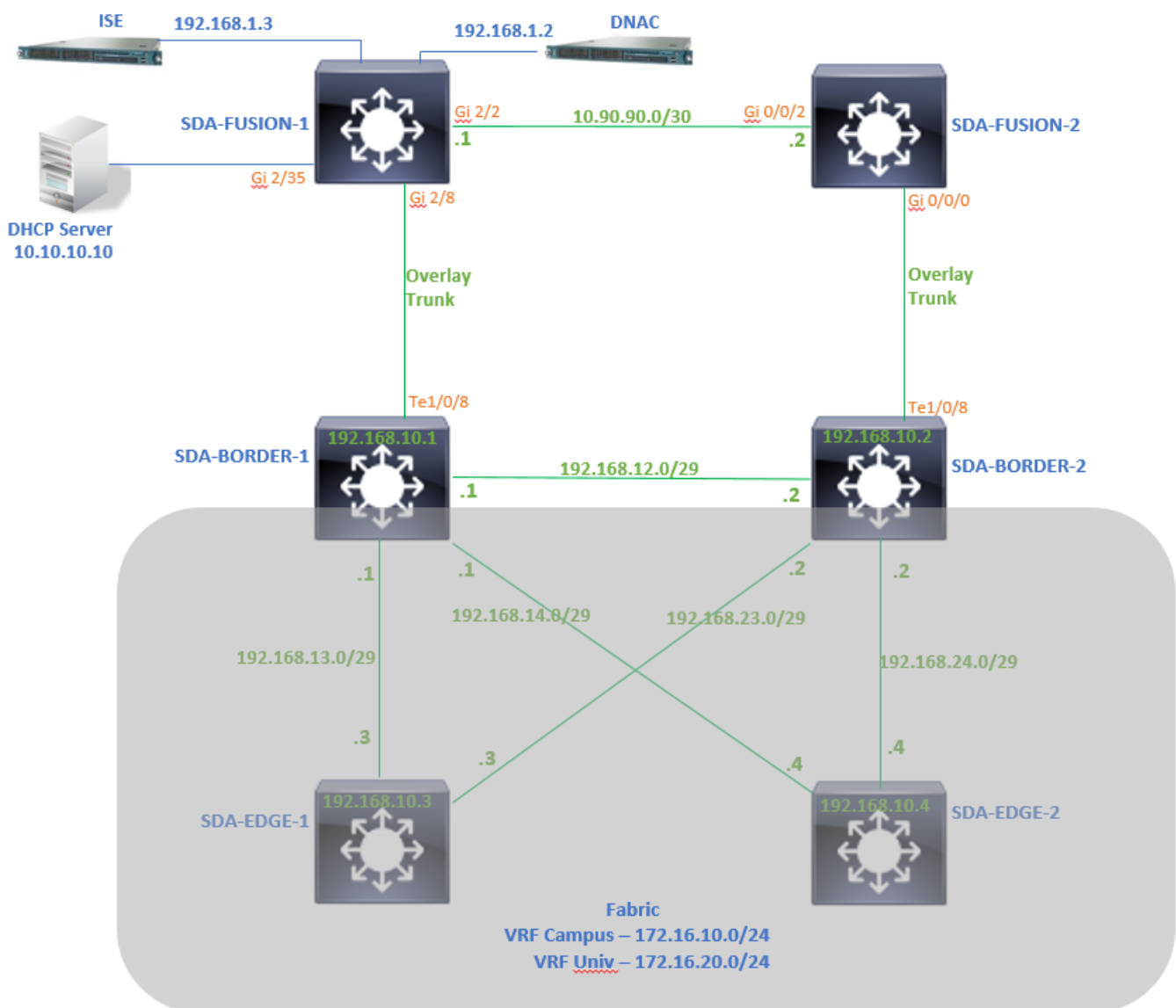
Opmerking: SD-Access border-knooppunten ondersteunen geen overzichtsroutes die overlappen met SD-Access IP-pools. Overzichtsroutes die overlappen met IP-pools moeten worden gefilterd in het routeren van advertenties van Fusion-apparaten naar border-knooppunten.

Configureren

De hier gegeven configuratiedetails zijn met betrekking tot de hieronder getoonde netwerktopologie. Deze netwerktopologie is geen aanbevolen topologie voor implementaties. Deze topologie wordt hier alleen gebruikt om de presentatie van de gegeven configuratievoorbeelden te vergemakkelijken. Raadpleeg [Design Zone for Cisco Digital Network Architecture](#) (Designzone voor digitale netwerkarchitectuur van Cisco) voor de aanbevolen implementatieontwerpen.

Netwerkdigram

De topologie die wordt gebruikt voor dit artikel bestaat uit twee border-routers die beide zijn geconfigureerd als externe borders en twee fusierouters die elk zijn verbonden met een border-router.



Configuraties

Stap 1. De koppeling Hand-off vanuit DNAC configureren

Binnen de stap van het toewijzen van apparaten een rol van Border Router terwijl het wordt toegevoegd aan de Fabric, kan een hand-off link worden gecreëerd. Bij Layer 2 is het een trunklink die is aangesloten op de Fusion Router. De volgende stappen zijn noodzakelijk:

1. Configureer een lokaal AS-nummer voor BGP. Dit AS-nummer (autonoom systeem) wordt gebruikt om het BGP-proces op de borderrouters te configureren.
2. Voeg een interface toe onder Transit. Deze interface is de directe verbinding tussen border- en fusierouter. (Te 1/0/8 op border in dit voorbeeld).

SDA-Border1

Border to

- Rest of Company (Internal)
- Outside World (External)
- Anywhere (Internal & External)

Local Autonomous Number

65005



Select Ip Pool

✖ BGP (10.50.50.0/24)



Connected to the Internet

Transit



Add

▼ ABC



External Interface

Add Interface

Interface

Number of VN

TenGigabitEthernet1/0/8

2

3. Configureer het nummer van het externe AS. Dit AS-nummer wordt gebruikt op border-routers voor buurverklaringen naar Fusion Router om externe BGP-peers (eBGP) te configureren.
4. Selecteer alle virtuele netwerken (VRF's) waarvoor VRF-doorgifte vereist is op de fusierouter.
5. Implementeer de configuratie van DNAC naar apparaten.

SDA-Border1

[< Back](#)

External Interface

* TenGigabitEthernet1/0/8

Remote AS Number

65004



This number is automatically derived from the selected Transit.
The selected autonomous system number will be used to automate IP routing between Border Node and remote peer.

Virtual Network

DEFAULT_VN

INFRA_VN

Univ

Campus

Gebruik dezelfde stappen voor het SDA-Border-2 apparaat.

Stap 2. Controleer de configuraties die door DNAC op border-routers worden uitgevoerd

In deze sectie wordt de verificatie van de configuratie op border-routers met betrekking tot het BGP-protocol behandeld.

SDA-Border-1

```
SDA-Border1#show run interface loopback 0
!
interface Loopback0
ip address 192.168.10.1 255.255.255.255
ip router isis
end
```

```
SDA-Border1#show run interface tenGigabitEthernet 1/0/8
!
interface TenGigabitEthernet1/0/8
switchport mode trunk
end
```

```
SDA-Border1#show run interface loopback 1021

interface Loopback1021
description Loopback Border
vrf forwarding Campus
ip address 172.16.10.1 255.255.255.255
end
```

```
SDA-Border1#show run interface loopback 1022
```

```
interface Loopback1022
description Loopback Border
vrf forwarding Univ
ip address 172.16.20.1 255.255.255.255
end
```

```
SDA-Border1#show run | section vrf definition Campus
vrf definition Campus
rd 1:4099
!
address-family ipv4
route-target export 1:4099
route-target import 1:4099
exit-address-family
```

```
SDA-Border1#show run | section vrf definition Univ
vrf definition Univ
rd 1:4100
!
address-family ipv4
route-target export 1:4100
route-target import 1:4100
exit-address-family
SDA-Border1#
```

```
SDA-Border1#show run interface vlan 3007
!
interface Vlan3007 <<< SVI created for BGP Peering under VRF Campus
description vrf interface to External router
vrf forwarding Campus
ip address 10.50.50.25 255.255.255.252
no ip redirects
ip route-cache same-interface
end
```

```
SDA-Border1#show run interface vlan 3006
!
interface Vlan3006 <<< SVI created for BGP Peering under VRF Univ
description vrf interface to External router
vrf forwarding Univ
ip address 10.50.50.21 255.255.255.252
no ip redirects
ip route-cache same-interface
end
```

```
SDA-Border1#show run | section bgp
router bgp 65005 <<< Local AS Number from DNAC
bgp router-id interface Loopback0
bgp log-neighbor-changes
bgp graceful-restart
!
address-family ipv4
network 192.168.10.1 mask 255.255.255.255
redistribute lisp metric 10
exit-address-family
!
address-family ipv4 vrf Campus
bgp aggregate-timer 0
network 172.16.10.1 mask 255.255.255.255 <<< Anycast IP for Pool in VRF Campus
aggregate-address 172.16.10.0 255.255.255.0 summary-only <<< Only Summary is Advertised
```

```

redistribute lisp metric 10
neighbor 10.50.50.26 remote-as 65004 <<< Peer IP to be used on Fusion for VRF Campus and Remote
AS Number from DNAC
neighbor 10.50.50.26 update-source Vlan3007
neighbor 10.50.50.26 activate
neighbor 10.50.50.26 weight 65535 <<< Weight needed for Fusion peering to make sure locally
originated path from LISP is never preferred
exit-address-family
!
address-family ipv4 vrf Univ
bgp aggregate-timer 0
network 172.16.20.1 mask 255.255.255.255 <<< Anycast IP for Pool in VRF Univ
aggregate-address 172.16.20.0 255.255.255.0 summary-only
redistribute lisp metric 10
neighbor 10.50.50.22 remote-as 65004
neighbor 10.50.50.22 update-source Vlan3006
neighbor 10.50.50.22 activate
neighbor 10.50.50.22 weight 65535
exit-address-family

```

SDA-Border-2

```

SDA-Border2#show run interface loopback 0
!
interface Loopback0
 ip address 192.168.10.2 255.255.255.255
 ip router isis
end

```

```

SDA-Border2#show run interface tenGigabitEthernet 1/0/8
!
interface TenGigabitEthernet1/0/8
 switchport mode trunk
end

```

```

SDA-Border2#show run interface loopback 1021
!
interface Loopback1021
 description Loopback Border
 vrf forwarding Campus
 ip address 172.16.10.1 255.255.255.255
end

```

```

SDA-Border2#show run interface loopback 1022
!
interface Loopback1022
 description Loopback Border
 vrf forwarding Univ
 ip address 172.16.20.1 255.255.255.255
end

```

```

SDA-Border2#show run | section vrf definition Campus vrf definition Campus rd 1:4099 ! address-
family ipv4 route-target export 1:4099 route-target import 1:4099 exit-address-family SDA-
Border2#show run | section vrf definition Univ vrf definition Univ rd 1:4100 ! address-family
ipv4 route-target export 1:4100 route-target import 1:4100 exit-address-family SDA-Border2#show
run interface vlan 3001 ! interface Vlan3001 description vrf interface to External router vrf
forwarding Campus ip address 10.50.50.1 255.255.255.252 no ip redirects ip route-cache same-
interface end SDA-Border2#show run interface vlan 3003 ! interface Vlan3003 description vrf
interface to External router vrf forwarding Univ ip address 10.50.50.9 255.255.255.252 no ip
redirects ip route-cache same-interface end SDA-Border2#show run | section bgp router bgp 65005
bgp router-id interface Loopback0 bgp log-neighbor-changes bgp graceful-restart ! address-family

```

```
ipv4 network 192.168.10.2 mask 255.255.255.255 redistribute lisp metric 10 exit-address-family !
address-family ipv4 vrf Campus bgp aggregate-timer 0 network 172.16.10.1 mask 255.255.255.255
aggregate-address 172.16.10.0 255.255.255.0 summary-only redistribute lisp metric 10 neighbor
10.50.50.2 remote-as 65004 neighbor 10.50.50.2 update-source Vlan3001 neighbor 10.50.50.2
activate neighbor 10.50.50.2 weight 65535 exit-address-family ! address-family ipv4 vrf Univ bgp
aggregate-timer 0 network 172.16.20.1 mask 255.255.255.255 aggregate-address 172.16.20.0
255.255.255.0 summary-only redistribute lisp metric 10 neighbor 10.50.50.10 remote-as 65004
neighbor 10.50.50.10 update-source Vlan3003 neighbor 10.50.50.10 activate neighbor 10.50.50.10
weight 65535 exit-address-family
```

Stap 3. Inloggen op border-routers configureren

Door de VRF-doorgifte op de fusierouter leert address-family ipv4 vrf Campus de route afkomstig uit vrf Univ (172.16.20.0/24). De bronrouter en de lerende router hebben echter hetzelfde BGP AS-nummer (65005). Om mechanismen voor het voorkomen van BGP-lussen te overwinnen en de routes op border-routers te accepteren en te installeren, moet **allowas-in** worden geconfigureerd voor de peerings met de fusierouter:

SDA-Border1

```
SDA-Border1(config)#router bgp 65005
SDA-Border1(config-router)#address-family ipv4 vrf Campus
SDA-Border1(config-router-af)#neighbor 10.50.50.26 allowas-in
SDA-Border1(config-router-af)#exit-address-family
SDA-Border1(config-router)#
SDA-Border1(config-router)#address-family ipv4 vrf Univ
SDA-Border1(config-router-af)#neighbor 10.50.50.22 allowas-in
SDA-Border1(config-router-af)#exit-address-family
SDA-Border1(config-router)#
```

SDA-Border2

```
SDA-Border2(config)#router bgp 65005
SDA-Border2(config-router)#address-family ipv4 vrf Campus
SDA-Border2(config-router-af)#neighbor 10.50.50.2 allowas-in
SDA-Border2(config-router-af)#exit-address-family
SDA-Border2(config-router)#
SDA-Border2(config-router)#address-family ipv4 vrf Univ
SDA-Border2(config-router-af)#neighbor 10.50.50.10 allowas-in
SDA-Border2(config-router-af)#exit-address-family
SDA-Border2(config-router)#
```

Opmerking: Opdracht **allowas-in** moet worden gebruikt met voorzorg omdat het kan leiden tot loops. Als u slechts één Fusion-apparaat gebruikt waar beide Borders mee werken, is filtering nodig om ervoor te zorgen dat lokaal voortgekomen routes niet worden geaccepteerd in het AS van de Fusion-peer - binnen dezelfde VN. Als dat gebeurt, krijgt het eBGP-pad vanwege het maximale gewicht voor eBGP-paden de voorkeur boven het lokaal gegenereerde pad.

Stap 4. Fusion-routers configureren

In deze sectie wordt de handmatige configuratie voor de fusierouters beschreven.

SDA-Fusion-1

Configureer de link naar de border-router als een trunk om deze af te stemmen met de VLAN-configuratie van Border-1:

```
interface GigabitEthernet2/8
  switchport
  switchport trunk encapsulation dot1q
  switchport trunk allowed vlan 3006, 3007
  switchport mode trunk
end
```

Configureer de vereiste VRF's:

```
vrf definition Campus
  rd 1:4099
  !
  address-family ipv4
    route-target export 1:4099
    route-target import 1:4099
  exit-address-family
!
```

```
vrf definition Univ
  rd 1:4100
  !
  address-family ipv4
    route-target export 1:4100
    route-target import 1:4100
  exit-address-family
```

Configureer SVI-interfaces:

```
interface Vlan3007
  vrf forwarding Campus
  ip address 10.50.50.26 255.255.255.252
end
```

```
interface Vlan3006
  vrf forwarding Univ
  ip address 10.50.50.22 255.255.255.252
end
```

Configureer eBGP-peering (externe BGP) met SDA-Border-1:

```
router bgp 65004                                     <<< Remote AS from DNAC
  bgp log-neighbor-changes
  !
  address-family ipv4
  exit-address-family
  !
  address-family ipv4 vrf Campus
    neighbor 10.50.50.25 remote-as 65005
    neighbor 10.50.50.25 update-source Vlan3007
    neighbor 10.50.50.25 activate
  exit-address-family
!
```

```
address-family ipv4 vrf Univ
  neighbor 10.50.50.21 remote-as 65005
  neighbor 10.50.50.21 update-source Vlan3006
  neighbor 10.50.50.21 activate
exit-address-family
```

Configureer iBGP-peering (interne BGP) met SDA-Fusion-2:

```
interface GigabitEthernet2/2
  description SDA-Fusion1--->SDA-Fusion2
  ip address 10.90.90.1 255.255.255.252
end
```

```
router bgp 65004
  neighbor 10.90.90.2 remote-as 65004
  !
  address-family ipv4
    neighbor 10.90.90.2 activate
  exit-address-family
  !
```

Geef het DHCP-serversubnet door onder de globale adresfamilie waar het IP-adres van de DHCP-server 10.10.10.10 is:

```
interface GigabitEthernet2/35
  description connection to DHCP server
  ip address 10.10.10.9 255.255.255.252
end
```

```
router bgp 65004
  !
  address-family ipv4
    network 10.10.10.8 mask 255.255.255.252
  exit-address-family
  !
```

SDA-Fusion-2

Configureer de link naar de border-router. Als een fusie-interface L3 is in plaats van trunk, configureert u subinterfaces:

```
interface GigabitEthernet0/0/0.3001
  encapsulation dot1Q 3001
  vrf forwarding Campus
  ip address 10.50.50.2 255.255.255.252
end
```

```
interface GigabitEthernet0/0/0.3003
  encapsulation dot1Q 3003
  vrf forwarding Univ
  ip address 10.50.50.10 255.255.255.252
end
```

Configureer de bijbehorende VRF's:

```

vrf definition Campus
 rd 1:4099
 !
 address-family ipv4
  route-target export 1:4099
  route-target import 1:4099
 exit-address-family
!
!
vrf definition Univ
 rd 1:4100
 !
 address-family ipv4
  route-target export 1:4100
  route-target import 1:4100
 exit-address-family
!

```

Configureer eBGP-peering met SDA-Border-2:

```

router bgp 65004
 bgp log-neighbor-changes
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv4 vrf Campus
  neighbor 10.50.50.1 remote-as 65005
  neighbor 10.50.50.1 update-source GigabitEthernet0/0/0.3001
  neighbor 10.50.50.1 activate
 exit-address-family
 !
 address-family ipv4 vrf Univ
  neighbor 10.50.50.9 remote-as 65005
  neighbor 10.50.50.9 update-source GigabitEthernet0/0/0.3003
  neighbor 10.50.50.9 activate
 exit-address-family

```

Configureer iBGP-peering met SDA-Fusion-1:

```

interface GigabitEthernet0/0/2
 ip address 10.90.90.2 255.255.255.252
 negotiation auto
 end

```

```

router bgp 65004 neighbor 10.90.90.1 remote-as 65004 ! address-family ipv4 neighbor 10.90.90.1
 activate exit-address-family

```

Stap 5. VRF-lekkage op fusierouterie configureren

De configuratie voor VRF-doorgifte is identiek voor de fusierouters SDA-Fusion-1 en SDA-Fusion-2.

Configureer eerst de VRF-lekkage tussen de twee VRF's (Campus en Univ) en gebruik **route-target import**:

```

vrf definition Campus
!
address-family ipv4
route-target export 1:4099 route-target import 1:4099
route-target import 1:4100 <<< Import VRF Univ prefixes in VRF Campus
exit-address-family
!
vrf definition Univ
!
address-family ipv4
route-target export 1:4100 route-target import 1:4100
route-target import 1:4099 <<< Import VRF Campus prefixes in VRF Univ
exit-address-family
!

```

Configureer vervolgens de route die lekt tussen de Global Routing Table (GRT) naar de VRF's en van de VRF's naar de GRT, gebruik **import ... map** en **export ... map**:

```

ip prefix-list Campus_Prefix seq 5 permit 172.16.10.0/24 <<< Include Prefixes belonging to
VRF Campus
ip prefix-list Global_Prefix seq 5 permit 10.10.10.8/30 <<< Include Prefixes belonging to
Global (eq DHCP Server Subnet)
ip prefix-list Univ_Prefix seq 5 permit 172.16.20.0/24 <<< Include Prefixes belonging to
VRF Univ

route-map Univ_Map permit 10
match ip address prefix-list Univ_Prefix
route-map Global_Map permit 10
match ip address prefix-list Global_Prefix
route-map Campus_Map permit 10
match ip address prefix-list Campus_Prefix

```

```

vrf definition Campus
!
address-family ipv4
import ipv4 unicast map Global_Map <<< Injecting Global into VRF Campus matching route-map
Global_Map
export ipv4 unicast map Campus_Map <<< Injecting VRF Campus into Global matching route-map
Campus_Map
exit-address-family
!
vrf definition Univ
!
address-family ipv4
import ipv4 unicast map Global_Map <<< Injecting Global into VRF Univ matching route-map
Global_Map
export ipv4 unicast map Univ_Map <<< Injecting VRF Univ into Global matching route-map Univ_Map
exit-address-family
!

```

Verifiëren

Deze sectie bevat controlestappen om er zeker van te zijn dat de vorige configuratie correct is uitgevoerd.

Stap 1. Controleer de eBGP-peer tussen fusie- en grensrouters

SDA-Border-1 -----Peering-----SDA-Fusion-1

SDA-Border1#show ip bgp vpnv4 vrf Campus summary

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
10.50.50.26	4	65004	1294	1295	32	0	0	19:32:22	2

SDA-Border1#show ip bgp vpnv4 vrf Univ summary

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
10.50.50.22	4	65004	1294	1292	32	0	0	19:32:57	2

SDA-Fusion1#show ip bgp vpnv4 vrf Campus summary

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
10.50.50.25	4	65005	1305	1305	31	0	0	19:41:58	1

SDA-Fusion1#show ip bgp vpnv4 vrf Univ summary

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
10.50.50.21	4	65005	1303	1305	31	0	0	19:42:14	1

SDA-Border-2 -----Peering-----SDA-Fusion-2

SDA-Border2#show ip bgp vpnv4 vrf Campus summary

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
10.50.50.2	4	65004	6	6	61	0	0	00:01:37	2

SDA-Border2#show ip bgp vpnv4 vrf Univ summary

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
10.50.50.10	4	65004	6	6	61	0	0	00:01:39	2

SDA-Fusion2#show ip bgp vpnv4 vrf Campus summary

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
10.50.50.1	4	65005	17	17	9	0	0	00:11:16	1

SDA-Fusion2#show ip bgp vpnv4 vrf Univ summary

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
10.50.50.9	4	65005	17	17	9	0	0	00:11:33	1

Stap 2. Controleer de iBGP-peer tussen beide Fusion-routers

SDA-Fusion-1 -----Peering-----SDA-Fusion-2

SDA-Fusion1#show ip bgp summary

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
10.90.90.2	4	65004	10	12	12	0	0	00:04:57	2

SDA-Fusion2#show ip bgp summary

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
10.90.90.1	4	65004	19	17	4	0	0	00:11:35	3

Stap 3. Controleer prefixes in BGP-tabel en routingtabel

SDA-Border-1

SDA-Border1#show ip bgp vpnv4 vrf Campus

Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 1:4099 (default for vrf Campus)					
*> 10.10.10.8/30	10.50.50.26	65535	65004	i	<<< Prefix leaked from Global Routing Table on Fusion
*> 172.16.10.0/24	0.0.0.0	32768	i		<<< VRF Campus originated prefix
*> 172.16.20.0/24	10.50.50.26	65535	65004	65005	i <<< Prefix originated in VRF Univ, leaked on Fusion to VRF Campus

SDA-Border1#show ip route vrf Campus bgp Routing Table: Campus B 10.10.10.8/30 [20/0] via 10.50.50.26, 20:30:30 <<< RIB entry for DHCP Server pool prefix B 172.16.10.0/24 [200/0], 20:32:45, Null0 <<< Null entry created by "aggregate-address" BGP configuration B 172.16.20.0/24 [20/0] via 10.50.50.26, 20:32:45 <<< RIB entry for VRF Univ prefix -----

----- SDA-Border1#show ip bgp vpnv4 vrf Univ Network

Next Hop	Metric	LocPrf	Weight	Path	Route Distinguisher: 1:4100 (default for vrf Univ) *
10.10.10.8/30	10.50.50.22	65535	65004	i	<<< Prefix leaked from Global Routing Table on Fusion *
172.16.10.0/24	10.50.50.22	65535	65004	65005	i <<< Prefix originated in VRF Campus, leaked on Fusion to VRF Univ *
172.16.20.0/24	0.0.0.0	32768	i		<<< VRF Univ originated prefix

SDA-Border1#show ip route vrf Univ bgp Routing Table: Univ B 10.10.10.8/30 [20/0] via 10.50.50.22, 20:31:06 <<< RIB entry for DHCP Server pool prefix B 172.16.10.0/24 [20/0] via 10.50.50.22, 20:33:21 <<< RIB entry for VRF Campus prefix B 172.16.20.0/24 [200/0], 20:33:21, Null0 <<< Null entry created by "aggregate-address" BGP configuration

SDA-Border-2

SDA-Border2#show ip bgp vpnv4 vrf Campus

Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 1:4099 (default for vrf Campus)					
*> 10.10.10.8/30	10.50.50.2	65535	65004	i	<<< Prefix leaked from Global Routing Table on Fusion
*> 172.16.10.0/24	0.0.0.0	32768	i		<<< VRF Campus originated prefix
*> 172.16.20.0/24	10.50.50.2	65535	65004	65005	i <<< Prefix originated in VRF Univ, leaked on Fusion to VRF Campus

SDA-Border2#show ip route vrf Campus bgp

B 10.10.10.8/30 [20/0] via 10.50.50.2, 01:02:19 <<< RIB entry for DHCP Server pool prefix

B 172.16.10.0/24 [200/0], 1w6d, Null0 <<< Null entry created by "aggregate-address" BGP configuration

B 172.16.20.0/24 [20/0] via 10.50.50.2, 01:02:27 <<< RIB entry for VRF Univ
Prefix

SDA-Border2#show ip bgp vpnv4 vrf Univ

Network	Next Hop	Metric	LocPrf	Weight	Path	
Route Distinguisher: 1:4100 (default for vrf Univ)						
*> 10.10.10.8/30	10.50.50.10	65535		65004	i	<<< Prefix
leaked from Global Routing Table on Fusion						
*> 172.16.10.0/24	10.50.50.10	65535		65004	65005 i	<<< Prefix
originated in VRF Campus, leaked on Fusion to VRF Univ						
*> 172.16.20.0/24	0.0.0.0	32768			i	<<< VRF Univ
originated prefix						

SDA-Border2#show ip route vrf Univ bgp

B 10.10.10.8/30 [20/0] via 10.50.50.10, 01:02:29 <<< RIB entry for DHCP Server
pool prefix
B 172.16.10.0/24 [20/0] via 10.50.50.10, 01:02:34 <<< RIB entry for VRF Campus
prefix
B 172.16.20.0/24 [200/0], 1w6d, Null0 <<< Null entry created by
"aggregate-address" BGP configuration

SDA-Fusion-1

SDA-Fusion1#show ip bgp

Network	Next Hop	Metric	LocPrf	Weight	Path	
*> 10.10.10.8/30	0.0.0.0	0		32768	i	<<< Locally
originated Global prefix						
* i 172.16.10.0/24	10.50.50.1	0	100	0	65005 i	<<< Prefix imported
from VRF Campus						
*>	10.50.50.25	0		0	65005 i	
* i 172.16.20.0/24	10.50.50.9	0	100	0	65005 i	<<< Prefix imported
from VRF Univ						
*>	10.50.50.21	0		0	65005 i	

SDA-Fusion1#show ip route

C 10.10.10.8/30 is directly connected, GigabitEthernet2/35 <<< Prefix for DHCP
Server
B 172.16.10.0 [20/0] via 10.50.50.25 (Campus), 20:50:21 <<< Prefix imported
from VRF Campus
B 172.16.20.0 [20/0] via 10.50.50.21 (Univ), 20:50:21 <<< Prefix imported from
VRF Univ

SDA-Fusion1#show ip bgp vpnv4 vrf Campus

Network	Next Hop	Metric	LocPrf	Weight	Path	
Route Distinguisher: 1:4099 (default for vrf Campus)						
Import Map: Global_Map, Address-Family: IPv4 Unicast, Pfx Count/Limit: 1/1000						
Export Map: Campus_Map, Address-Family: IPv4 Unicast, Pfx Count/Limit: 1/1000						
*> 10.10.10.8/30	0.0.0.0	0		32768	i	<<< Prefix imported

```

from Global Routing
*> 172.16.10.0/24 10.50.50.25 0 0 65005 i <<< Prefix learnt from
Border1 in VRF Campus
*> 172.16.20.0/24 10.50.50.21 0 0 65005 i <<< Prefix imported from
VRF Univ

```

```

SDA-Fusion1#show ip bgp vpnv4 vrf Campus 172.16.20.0/24
BGP routing table entry for 1:4099:172.16.20.0/24, version 27
Paths: (1 available, best #1, table Campus)
Advertised to update-groups:
5
Refresh Epoch 1
65005, (aggregated by 65005 192.168.10.1), imported path from 1:4100:172.16.20.0/24 (Univ)
10.50.50.21 (via vrf Univ) (via Univ) from 10.50.50.21 (192.168.10.1)
Origin IGP, metric 0, localpref 100, valid, external, atomic-aggregate, best
Extended Community: RT:1:4100
rx pathid: 0, tx pathid: 0x0

```

```

SDA-Fusion1#show ip route vrf Campus bgp B 10.10.10.8/30 is directly connected, 20:46:51,
GigabitEthernet2/35 B 172.16.10.0 [20/0] via 10.50.50.25, 20:50:07 B 172.16.20.0 [20/0] via
10.50.50.21 (Univ), 20:50:07 -----
----- SDA-Fusion1#show ip bgp vpnv4 vrf Univ Network Next Hop Metric LocPrf Weight Path
Route Distinguisher: 1:4100 (default for vrf Univ) Import Map: Global_Map, Address-Family: IPv4
Unicast, Pfx Count/Limit: 1/1000 Export Map: Univ_Map, Address-Family: IPv4 Unicast, Pfx
Count/Limit: 1/1000 *> 10.10.10.8/30 0.0.0.0 0 32768 i <<< Prefix imported from Global Routing
*> 172.16.10.0/24 10.50.50.25 0 0 65005 i <<< Prefix imported from VRF Campus *> 172.16.20.0/24
10.50.50.21 0 0 65005 i <<< Prefix learnt from Border1 in VRF Univ

```

```

SDA-Fusion1#show ip bgp vpnv4 vrf Univ 172.16.10.0/24
BGP routing table entry for 1:4100:172.16.10.0/24, version 25
Paths: (1 available, best #1, table Univ)
Advertised to update-groups:
4
Refresh Epoch 1
65005, (aggregated by 65005 192.168.10.1), imported path from 1:4099:172.16.10.0/24 (Campus)
10.50.50.25 (via vrf Campus) (via Campus) from 10.50.50.25 (192.168.10.1)
Origin IGP, metric 0, localpref 100, valid, external, atomic-aggregate, best
Extended Community: RT:1:4099
rx pathid: 0, tx pathid: 0x0

```

```

SDA-Fusion1#show ip route vrf Univ bgp B 10.10.10.8/30 is directly connected, 20:47:01,
GigabitEthernet2/35 B 172.16.10.0 [20/0] via 10.50.50.25 (Campus), 20:50:17 B 172.16.20.0 [20/0]
via 10.50.50.21, 20:50:17

```

SDA-Fusion-2

```

SDA-Fusion2#show ip bgp

```

	Network	Next Hop	Metric	LocPrf	Weight	Path
*>i	10.10.10.8/30	10.90.90.1	0	100	0	i
*>	172.16.10.0/24	10.50.50.1	0		0	65005 i
* i		10.50.50.25	0	100	0	65005 i
*>	172.16.20.0/24	10.50.50.9	0		0	65005 i
* i		10.50.50.21	0	100	0	65005 i

```

SDA-Fusion2#show ip route

```

```

B 10.10.10.8/30 [200/0] via 10.90.90.1, 01:25:56
B 172.16.10.0 [20/0] via 10.50.50.1 (Campus), 01:25:56

```



```
B      172.16.20.0 [20/0] via 10.50.50.9 (Univ), 01:25:56
```

```
-----  
SDA-Fusion2#show ip bgp vpnv4 vrf Campus
```

```
      Network          Next Hop          Metric LocPrf Weight Path  
Route Distinguisher: 1:4099 (default for vrf Campus)  
Import Map: Global_Map, Address-Family: IPv4 Unicast, Pfx Count/Limit: 1/1000  
Export Map: Campus_Map, Address-Family: IPv4 Unicast, Pfx Count/Limit: 1/1000  
*>i 10.10.10.8/30      10.90.90.1          0    100    0 i  
*> 172.16.10.0/24     10.50.50.1          0          0 65005 i  
*> 172.16.20.0/24     10.50.50.9          0          0 65005 i
```

```
SDA-Fusion2#show ip route vrf Campus bgp
```

```
B      10.10.10.8/30 [200/0] via 10.90.90.1, 01:26:09  
B      172.16.10.0 [20/0] via 10.50.50.1, 01:26:13  
B      172.16.20.0 [20/0] via 10.50.50.9 (Univ), 01:26:13
```

```
-----  
SDA-Fusion2#show ip bgp vpnv4 vrf Univ
```

```
      Network          Next Hop          Metric LocPrf Weight Path  
Route Distinguisher: 1:4100 (default for vrf Univ)  
Import Map: Global_Map, Address-Family: IPv4 Unicast, Pfx Count/Limit: 1/1000  
Export Map: Univ_Map, Address-Family: IPv4 Unicast, Pfx Count/Limit: 1/1000  
*>i 10.10.10.8/30      10.90.90.1          0    100    0 i  
*> 172.16.10.0/24     10.50.50.1          0          0 65005 i  
*> 172.16.20.0/24     10.50.50.9          0          0 65005 i
```

```
SDA-Fusion2#show ip route vrf Univ bgp
```

```
B      10.10.10.8/30 [200/0] via 10.90.90.1, 01:26:19  
B      172.16.10.0 [20/0] via 10.50.50.1 (Campus), 01:26:23  
B      172.16.20.0 [20/0] via 10.50.50.9, 01:26:23
```

Handmatige configuratie voor border-redundantie

Voor redundantie tussen de PETR's wanneer een externe link aan de grens mislukt, moet u voor de externe en externe+interne grenzen iBGP-sessies tussen de twee randen voor elk van de VPN's handmatig bouwen. Bovendien zijn in het geval van de Externe+Interne grens waar BGP wordt ingevoerd in LISP en LISP terug in BGP wordt herverdeeld, markeringen nodig om iBGP te verhinderen aan LISP routeinvoer en zo potentiële loops te vermijden.

SDA-Border-1

```
interface Vlan31  
  description vrf interface to SDA-Border-2  
  vrf forwarding Campus  
  ip address 10.31.1.1 255.255.255.252  
!  
interface Vlan33
```

```

description vrf interface to SDA-Border-2
vrf forwarding Univ
ip address 10.33.1.1 255.255.255.252
!

router bgp 65005
!
address-family ipv4 vrf Campus
redistribute lisp metric 10 <<< open redistribution pushed by DNAC
neighbor 10.31.1.2 remote-as 65005 <<< iBGP peering with SDA-Border-2
neighbor 10.31.1.2 activate
neighbor 10.31.1.2 send-community <<< we need to send community/tag to the neighbor
neighbor 10.31.1.2 route-map tag_local_eids out <<< route-map used to tag prefixes sent out
!
address-family ipv4 vrf Univ
redistribute lisp metric 10
neighbor 10.33.1.2 remote-as 65005
neighbor 10.33.1.2 activate
neighbor 10.33.1.2 send-community
neighbor 10.33.1.2 route-map tag_local_eids out
!

router lisp
!
instance-id 4099
service ipv4
eid-table vrf Campus
route-import database bgp 65005 route-map DENY-Campus locator-set rloc_a0602921-91eb-4e27-a294-
f88949a1ca37 <<< pushed by DNAC if Border is (also) Internal
!
instance-id 4103
service ipv4
eid-table vrf Univ
route-import database bgp 65005 route-map DENY-Univ locator-set rloc_a0602921-91eb-4e27-a294-
f88949a1ca37
!

ip community-list 1 permit 655370 <<< community-list matching tag 655370 - pushed by DNAC
!

route-map DENY-Campus deny 5 <<< route-map pushed by DNAC and used in route-import
match ip address prefix-list Campus
!
route-map DENY-Campus deny 10
match ip address prefix-list l3handoff-prefixes
!
route-map DENY-Campus deny 15
match community 1 <<< match on community-list 1 to deny iBGP prefixes to be imported into LISP
!
route-map DENY-Campus deny 25
match ip address prefix-list deny_0.0.0.0
!
route-map DENY-Campus permit 30
!

route-map DENY-Univ deny 5 <<< similar route-map is pushed for Univ VN
match ip address prefix-list Univ
!
route-map DENY-Univ deny 10
match ip address prefix-list l3handoff-prefixes
!
route-map DENY-Univ deny 15
match community 1
!

```

```
route-map DENY-Univ deny 25
match ip address prefix-list deny_0.0.0.0
!
route-map DENY-Univ permit 30
!

route-map tag_local_eids permit 5 <<< route-map we need to create in order to tag the routes
advertised to the iBGP peer
set community 655370 <<< setting community/tag to 655370
!
```

SDA-Border-2

```
interface Vlan31
description vrf interface to SDA-Border-1
vrf forwarding Campus
ip address 10.31.1.2 255.255.255.252
!
interface Vlan33
description vrf interface to SDA-Border-1
vrf forwarding Univ
ip address 10.33.1.2 255.255.255.252
!

router bgp 65005
!
address-family ipv4 vrf Campus
neighbor 10.31.1.1 remote-as 65005
neighbor 10.31.1.1 activate
neighbor 10.31.1.1 send-community
neighbor 10.31.1.1 route-map tag_local_eids out
!
address-family ipv4 vrf Univ
neighbor 10.33.1.1 remote-as 65005
neighbor 10.33.1.1 activate
neighbor 10.33.1.1 send-community
neighbor 10.33.1.1 route-map tag_local_eids out
!

router lisp
!
instance-id 4099
service ipv4
eid-table vrf Campus
route-import database bgp 65005 route-map DENY-Campus locator-set rloc_677c0a8a-0802-49f9-99cc-
f9c6ebda80f3 <<< pushed by DNAC
!

instance-id 4103
service ipv4
eid-table vrf Univ
route-import database bgp 65005 route-map DENY-Univ locator-set rloc_677c0a8a-0802-49f9-99cc-
f9c6ebda80f3
!

ip community-list 1 permit 655370
!

route-map DENY-Campus deny 5
match ip address prefix-list Campus
!
route-map DENY-Campus deny 10
match ip address prefix-list l3handoff-prefixes
```

```

!
route-map DENY-Campus deny 15
match community 1
!
route-map DENY-Campus deny 25
match ip address prefix-list deny_0.0.0.0
!
route-map DENY-Campus permit 30
!

route-map DENY-Univ deny 5
match ip address prefix-list Univ
!
route-map DENY-Univ deny 10
match ip address prefix-list l3handoff-prefixes
!
route-map DENY-Univ deny 15
match community 1
!
route-map DENY-Univ deny 25
match ip address prefix-list deny_0.0.0.0
!
route-map DENY-Univ permit 30
!

route-map tag_local_eids permit 5
set community 655370
!

```

Vereenvoudig fusieconfiguratie met het gebruik van sjablonen

Deze sectie bevat configuratievoorbeelden voor fusiesjablonen om de configuratie te vereenvoudigen.

Vervolgens zijn de Variabelen die moeten worden gedefinieerd op basis van uw implementatieontwerp. In dit voorbeeld zijn de configuraties en VPN's gebaseerd op de vorige topologie die twee VPN's heeft, Campus en Univ.

Definitie van variabelen

```

interface_Fusion1: GigabitEthernet2/8
interface_Fusion2: GigabitEthernet0/0/0

```

```

Global_prefixes = 10.10.10.8/30

```

```

FUSION_BGP_AS = 65004
BORDER_BGP_AS = 65005

```

Voor VN1:

```

VN1 = Campus
Fusion1_VN1_VLAN = 3007
Fusion2_VN1_VLAN = 3001
VN1_prefixes = 172.16.10.0/24

Fusion1_VN1_IP = 10.50.50.26

Fusion1_VN1_MASK = 255.255.255.252

```

Fusion2_VN1_IP = 10.50.50.2

Fusion2_VN1_MASK = 255.255.255.252

VN1_RD = 4099

VN1_border1_neighbor_IP = 10.50.50.25

VN1_border2_neighbor_IP = 10.50.50.1

Voor VN2:

VN2 = Univ

Fusion1_VN2_VLAN = 3006

Fusion2_VN2_VLAN = 3003

VN2_prefixes = 172.16.20.0/24

Fusion1_VN2_IP = 10.50.50.22

Fusion1_VN2_MASK = 255.255.255.252

Fusion2_VN2_IP2 = 10.50.50.10

Fusion2_VN2_MASK = 255.255.255.252

VN2_RD = 4100

VN2_border1_neighbor_IP = 10.50.50.21

VN2_border2_neighbor_IP = 10.50.50.9

Voorbeeld van sjabloon

Fusie 1

```
interface $interface_Fusion1
switchport
switchport mode trunk
switchport trunk allowed vlan add $Fusion1_VN1_VLAN, $Fusion1_VN2_VLAN
!
vlan $Fusion1_VN1_VLAN
no shut
!
vlan $Fusion1_VN2_VLAN
no shut
!
vrf definition $VN1
rd 1:$VN1_RD
!
address-family ipv4
route-target export 1:$VN1_RD
route-target import 1:$VN1_RD
route-target import 1:$VN2_RD
exit-address-family
!
vrf definition $VN2
rd 1:$VN2_RD
!
address-family ipv4
route-target export 1:$VN2_RD
route-target import 1:$VN2_RD
route-target import 1:$VN1_RD
exit-address-family
!
interface Vlan $Fusion1_VN1_VLAN
vrf forwarding $VN1
ip address $Fusion1_VN1_IP $Fusion1_VN1_MASK
!
```

```

interface Vlan $Fusion1_VN2_VLAN
vrf forwarding $VN2
ip address $Fusion1_VN2_IP $Fusion1_VN2_MASK
!
router bgp $FUSION_BGP_AS
bgp log-neighbor-changes
!
address-family ipv4
exit-address-family
!
address-family ipv4 vrf $VN1
neighbor $VN1_border1_neighbor_IP remote-as $BORDER_BGP_AS
neighbor $VN1_border1_neighbor_IP update-source Vlan $Fusion1_VN1_VLAN
neighbor $VN1_border1_neighbor_IP activate
exit-address-family
!
address-family ipv4 vrf $VN2
neighbor $VN2_border1_neighbor_IP remote-as $BORDER_BGP_AS
neighbor $VN2_border1_neighbor_IP update-source $Fusion1_VN2_VLAN
neighbor $VN2_border1_neighbor_IP activate
exit-address-family

ip prefix-list ${VN1}_Prefix seq 5 permit $VN1_prefixes
ip prefix-list Global_Prefix seq 5 permit $Global_prefixes
ip prefix-list ${VN2}_Prefix seq 5 permit $VN2_prefixes

route-map ${VN2}_Map permit 10
match ip address prefix-list ${VN2}_Prefix
route-map Global_Map permit 10
match ip address prefix-list Global_Prefix
route-map ${VN1}_Map permit 10
match ip address prefix-list ${VN1}_Prefix

vrf definition $VN1
!
address-family ipv4
import ipv4 unicast map Global_Map
export ipv4 unicast map ${VN1}_Map
exit-address-family
!
vrf definition $VN2
!
address-family ipv4
import ipv4 unicast map Global_Map
export ipv4 unicast map ${VN2}_Map
exit-address-family
!

```

Fusie 2

```

interface $interface_Fusion2.$Fusion2_VN1_VLAN
encapsulation dot1Q $Fusion2_VN1_VLAN
vrf forwarding $VN1
ip address $Fusion2_VN1_IP2 $Fusion2_VN1_MASK
!
interface $interface_Fusion2.$Fusion2_VN2_VLAN
encapsulation dot1Q $Fusion2_VN2_VLAN
vrf forwarding $VN2
ip address $Fusion2_VN2_IP2 $Fusion2_VN2_MASK
!
vlan $Fusion2_VN1_VLAN
no shut

```

```

!
vlan $Fusion2_VN2_VLAN
no shut
!
vrf definition $VN1
rd 1:$VN1_RD
!
address-family ipv4
route-target export 1:$VN1_RD
route-target import 1:$VN1_RD
route-target import 1:$VN2_RD
exit-address-family
!
vrf definition $VN2
rd 1:$VN2_RD
!
address-family ipv4
route-target export 1:$VN2_RD
route-target import 1:$VN2_RD
route-target import 1:$VN1_RD
exit-address-family
!
router bgp $FUSION_BGP_AS
bgp log-neighbor-changes
!
address-family ipv4
exit-address-family
!
address-family ipv4 vrf $VN1
neighbor $VN1_border2_neighbor_IP remote-as $BORDER_BGP_AS
neighbor $VN1_border2_neighbor_IP update-source $interface_Fusion2.$Fusion2_VN1_VLAN
neighbor $VN1_bordre2_neighbor_IP activate
exit-address-family
!
address-family ipv4 vrf $VN2
neighbor $VN2_border2_neighbor_IP remote-as $BORDER_BGP_AS
neighbor $VN2_border2_neighbor_IP update-source $interface_Fusion2.$Fusion2_VN2_VLAN
neighbor $VN2_border2_neighbor_IP activate
exit-address-family

ip prefix-list ${VN1}_Prefix seq 5 permit $VN1_prefixes
ip prefix-list Global_Prefix seq 5 permit $Global_prefixes
ip prefix-list ${VN2}_Prefix seq 5 permit $VN2_prefixes

route-map ${VN2}_Map permit 10
match ip address prefix-list ${VN2}_Prefix
route-map Global_Map permit 10
match ip address prefix-list Global_Prefix
route-map ${VN}_Map permit 10
match ip address prefix-list ${VN1}_Prefix

vrf definition $VN1
!
address-family ipv4
import ipv4 unicast map Global_Map
export ipv4 unicast map ${VN1}_Map
exit-address-family
!
vrf definition $VN2
!
address-family ipv4
import ipv4 unicast map Global_Map
export ipv4 unicast map ${VN2}_Map
exit-address-family

```

!
End

Over deze vertaling

Cisco heeft dit document vertaald via een combinatie van machine- en menselijke technologie om onze gebruikers wereldwijd ondersteuningscontent te bieden in hun eigen taal. Houd er rekening mee dat zelfs de beste machinevertaling niet net zo nauwkeurig is als die van een professionele vertaler. Cisco Systems, Inc. is niet aansprakelijk voor de nauwkeurigheid van deze vertalingen en raadt aan altijd het oorspronkelijke Engelstalige document ([link](#)) te raadplegen.