

# SDA에 퓨전 라우터 구성

## 목차

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[배경 정보](#)

[DNA SD-Access 솔루션에서 Fusion 디바이스의 기능](#)

[구성](#)

[네트워크 다이어그램](#)

[설정](#)

[1단계. DNAC에서 핸드오프 링크 구성](#)

[2단계. 보더 라우터에서 DNAC에 의해 푸시된 컨피그레이션 확인](#)

[3단계. 보더 라우터에 allowas-in 구성](#)

[4단계. Fusion 라우터 구성](#)

[5단계. Fusion Router에서 VRF 유출 구성](#)

[다음을 확인합니다.](#)

[1단계. Fusion과 Border Router 간의 eBGP 피어링 확인](#)

[2단계. 두 Fusion 라우터 간의 iBGP 피어링 확인](#)

[3단계. BGP 테이블 및 라우팅 테이블의 접두사 확인](#)

[보더 이중화를 위한 수동 컨피그레이션](#)

[SDA-보더-1](#)

[SDA-보더-2](#)

[템플릿을 사용하여 Fusion 구성 간소화](#)

[변수 정의](#)

[템플릿 예](#)

[퓨전 1](#)

[퓨전 2](#)

## 소개

이 문서에서는 Cisco SDA(Software-Defined Access) 솔루션에서 Fusion Router를 구성하는 방법에 대해 설명합니다.

## 사전 요구 사항

### 요구 사항

이 문서에 대한 특정 요건이 없습니다.

참고: Link to release notes([릴리스 노트 링크](#))에서 찾을 수 있는 Supported Devices(지원되는 디바이스)에 따라 [설정이 필요합니다](#).

## 사용되는 구성 요소

이 문서의 내용은 다음 하드웨어 버전을 기반으로 합니다.

- DNAC - 버전 1.2.1
- 에지 및 경계 - Cat3k Cisco 스위치
- Fusion - VRF 간 유출을 지원하는 Cisco 라우터

이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 현재 네트워크가 작동 중인 경우 모든 명령의 잠재적인 영향을 미리 숙지하시기 바랍니다.

## 배경 정보

Cisco SD-Access 솔루션에서 디바이스는 Cisco DNA Center에서 관리하고 구성합니다. 일반적으로 SD-Access 패브릭의 모든 부분은 Cisco DNA Center에서 구성 및 관리할 수 있으며 일반적으로 구성 및 관리됩니다. 그러나 Fusion 디바이스는 패브릭 외부에 있으므로 수동으로 구성됩니다. 다음에 설명하는 Border Automation은 Cisco DNA Center 내의 기능으로, Fusion 디바이스에 VRF를 전달하기 위해 Border 컨피그레이션을 자동화할 수 있습니다.

경우에 따라 현재 컨피그레이션과의 호환성과 관련된 일반적인 이유로 Border Automation이 적합하지 않으므로 Border에서 Fusion 디바이스로의 핸드오프도 수동으로 구성할 수 있습니다. 사용된 구성에 대한 이해는 전체 시스템의 최적의 구성 및 운영에 대한 중요한 세부 사항을 설명하는 데 도움이 됩니다.

## DNA SD-Access 솔루션에서 Fusion 디바이스의 기능

Fusion 디바이스는 SD 액세스 패브릭 도메인 전반에 걸쳐 VRF(Virtual Routing and Forwarding)가 유출되도록 하며 DHCP, DNS, NTP, ISE, Cisco DNA Center, WLC(Wireless LAN Controller) 등과 같은 공유 서비스에 대한 호스트 연결을 지원합니다. 이 역할은 라우터가 아닌 다른 디바이스에서 수행할 수 있지만, 이 문서에서는 라우터를 Fusion 디바이스로 중점적으로 살펴봅니다.

앞서 언급한 대로 캠퍼스의 모든 VN(Virtual Network)에서 공유 서비스를 사용할 수 있어야 합니다. 이는 Border Router에서 Fusion Router로의 BGP(Border Gateway Protocol) 피어링을 생성하는 방법으로 구현됩니다. Fusion Router에서 이러한 공유 서비스에 액세스해야 하는 패브릭 VRF의 서브넷이 GRT 또는 공유 서비스 VRF로 유출되고 그 반대의 경우도 마찬가지입니다. 경로 맵은 SD-Access Fabric에 특정한 서브넷에 대한 라우팅 테이블을 포함하는 데 사용할 수 있습니다.

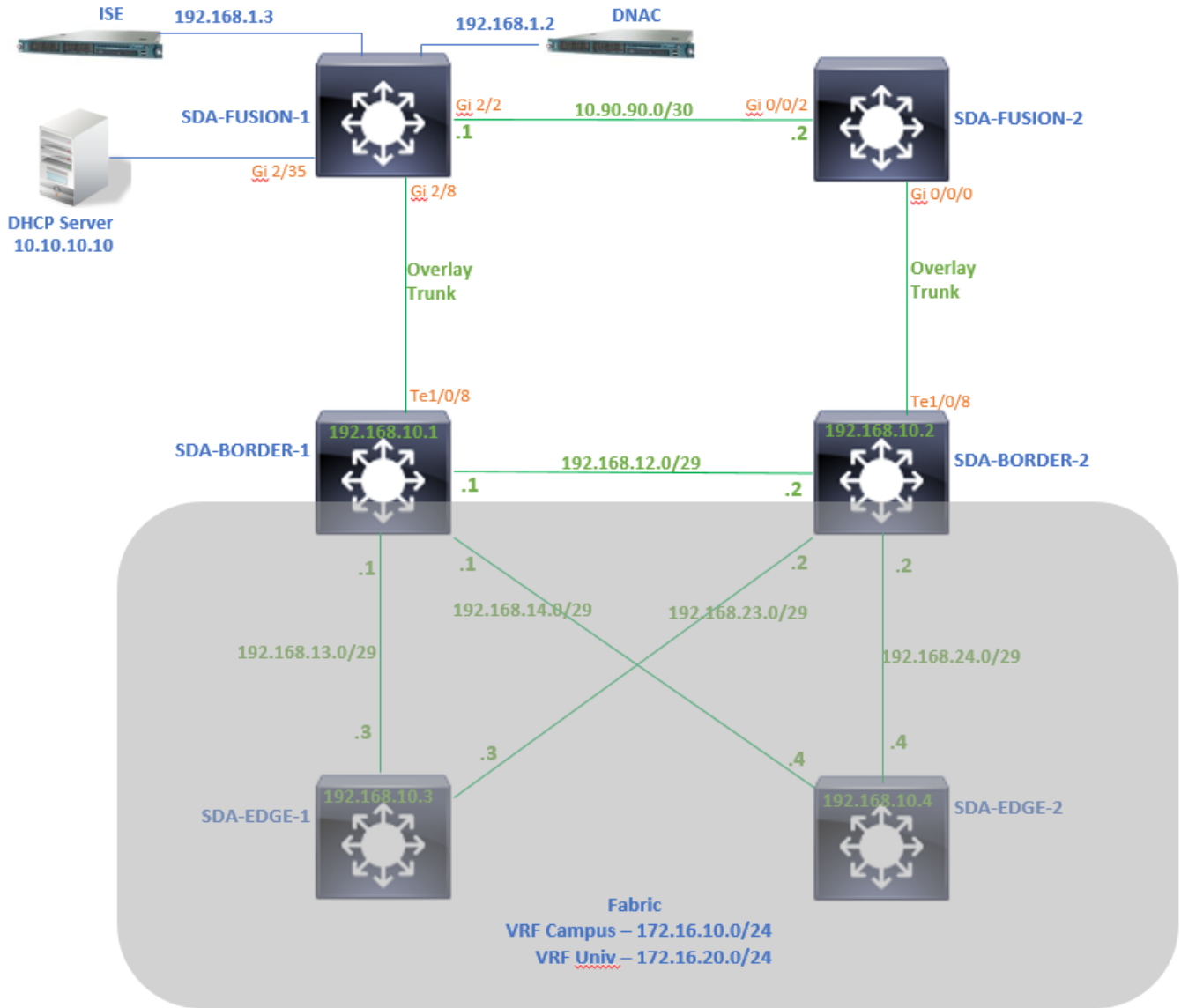
참고: SD-Access Border Node는 SD-Access IP 풀과 겹치는 요약 경로를 지원하지 않습니다. IP 풀과 겹치는 요약 경로는 Fusion 디바이스에서 Border Node로 광고를 라우팅할 때 필터링해야 합니다.

## 구성

여기에 제시된 컨피그레이션 세부사항은 다음에 표시된 네트워크 토폴로지와 관련된 것입니다. 이 네트워크 토폴로지는 구축에 권장되는 토폴로지가 아닙니다. 당해 물품은 여기에서 단지 제공된 구성 샘플의 프리젠테이션을 용이하게 하기 위해 사용됩니다. 권장 구축 설계는 [Cisco Digital Network Architecture의 Design Zone을 참조하십시오.](#)

# 네트워크 다이어그램

이 문서에 사용되는 토폴로지는 외부 경계로 구성된 두 개의 경계 라우터와 각 경계 라우터에 연결된 두 개의 Fusion 라우터로 구성됩니다.



## 설정

### 1단계. DNAC에서 핸드오프 링크 구성

디바이스에 Border Router 역할을 할당하는 단계에서 Fabric에 추가되면서 핸드오프 링크를 생성할 수 있습니다. 레이어 2에서는 Fusion Router에 연결된 트렁크 링크입니다. 다음 단계가 필요합니다.

1. BGP에 대한 로컬 AS 번호를 구성합니다. 이 AS(Autonomous System) 번호는 Border Router에서 BGP 프로세스를 구성하는 데 사용됩니다.
2. 전송 중인 인터페이스를 추가합니다. 이 인터페이스는 Border와 Fusion Router 간의 직접 연결입니다. (이 예에서는 Border(테두리)의 1/0/8입니다.)

## 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. 원격 AS 번호를 구성합니다. 이 AS 번호는 Border Routers에서 eBGP(External BGP) 피어를 구성하기 위해 Fusion Router에 대한 네이버 명령문에 사용됩니다.

4. Fusion Router에서 VRF 누수가 필요한 모든 가상 네트워크(VRF)를 선택합니다.

5. DNAC에서 디바이스로 컨피그레이션을 구축합니다.

## 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

SDA-Border-2 디바이스에 대해서도 동일한 단계를 수행합니다.

## 2단계. 보더 라우터에서 DNAC에 의해 푸시된 컨피그레이션 확인

이 섹션에서는 BGP 프로토콜과 관련된 보더 라우터의 컨피그레이션 확인에 대해 설명합니다.

### SDA-보더-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-보더-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
```

### 3단계. 보더 라우터에 allowas-in 구성

Fusion Router에서 VRF가 누출되므로 VRF 캠퍼스용 주소군 ipv4는 VRF Univ(172.16.20.0/24)에서 시작된 경로를 학습합니다. 발신 라우터와 학습 라우터 모두 동일한 BGP AS 번호(65005)를 갖지만 BGP 루프 방지 메커니즘을 극복하고 Border Router의 경로를 승인/설치하려면 Fusion Router와의 피어링에 대해 Allowas-in을 구성해야 합니다.

```
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)#
```

**참고:** 명령 승인 기능은 루프를 일으킬 수 있으므로 사전 예방과 함께 사용해야 합니다. 두 보더에서 모두 피어링하는 Fusion 디바이스를 하나만 사용하는 경우, 로컬에서 시작된 경로가 동일한 VN 내에서 Fusion 피어의 AS로 다시 받아들여지지 않도록 필터링해야 합니다. 이 경우 eBGP 경로에 대한 최대 가중치로 인해 eBGP 경로가 로컬에서 시작된 경로보다 우선합니다.

## 4단계. Fusion 라우터 구성

이 섹션에서는 Fusion 라우터의 수동 컨피그레이션을 설명합니다.

### SDA-퓨전-1

Border-1의 vlan 컨피그레이션과 일치하도록 보더 라우터에 대한 링크를 트렁크로 구성합니다.

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

필요한 VRF를 구성합니다.

```
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
```

SVI 인터페이스를 구성합니다.

```
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
```

SDA-Border-1을 사용하여 eBGP(external BGP) 피어링을 구성합니다.

```
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
```

SDA-Fusion-2를 사용하여 iBGP(internal BGP) 피어링을 구성합니다.

```
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
 !
```

DHCP 서버 IP가 10.10.10.10인 전역 주소군 아래에 DHCP 서버 서브넷을 광고합니다.

```
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-퓨전-2

Border Router에 대한 링크를 구성합니다. Fusion의 인터페이스가 트렁크가 아닌 L3인 경우 하위 인터페이스를 구성합니다.

```
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
```

해당 VRF를 구성합니다.

```
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
!
```

SDA-Border-2로 eBGP 피어링 구성:

```
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

```

## SDA-Fusion-1로 iBGP 피어링 구성:

```

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

```

## 5단계. Fusion Router에서 VRF 유출 구성

VRF 누출 구성은 Fusion Routers SDA-Fusion-1과 SDA-Fusion-2 모두에서 동일합니다.

먼저 두 VRF(Campus 및 Univ) 간 VRF 누설을 구성하고 경로 대상 가져오기를 사용합니다.

```

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
!

```

그런 다음 GRT(Global Routing Table)에서 VRF로, VRF에서 GRT로 경로 유출을 구성하려면 import ... map과 export.. map을 사용합니다.

```

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
!

```

## 다음을 확인합니다.

이 섹션에서는 이전 컨피그레이션이 올바르게 적용되었는지 확인하는 단계를 다룹니다.

### 1단계. Fusion과 Border Router 간의 eBGP 피어링 확인

#### SDA-Border-1-----피어링-----SDA-퓨전-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-----피어링-----SDA-퓨전-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

### 2단계. 두 Fusion 라우터 간의 iBGP 피어링 확인

SDA-Fusion-1-----피어링-----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

### 3단계. BGP 테이블 및 라우팅 테이블의 접두사 확인

SDA-보더-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-보더-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-퓨전-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-퓨전-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
```

## 보더 이중화를 위한 수동 컨피그레이션

경계 외부 링크가 실패할 때 PETR 간의 이중화를 위해 외부 및 외부+내부 경계의 경우 각 VN에 대해 두 경계 간에 iBGP 세션을 수동으로 구축해야 합니다. 또한 BGP를 LISP로 가져오고 LISP를 다시 BGP로 재배포하는 외부+내부 경계의 경우 iBGP에서 LISP 경로 가져오기를 방지하므로 잠재적 루프를 방지하기 위해 태그가 필요합니다.

### SDA-보더-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-f88949alca37 <<< 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-f88949alca37
!
```

```

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-보더-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
!

```

## 템플릿을 사용하여 Fusion 구성 간소화

이 섹션에서는 컨피그레이션을 간소화하는 데 도움이 되는 샘플 Fusion Template 컨피그레이션 예 를 제공합니다.

다음은 구축 설계에 따라 를 정의하는 데 필요한 변수입니다. 이 예에서 컨피그레이션 및 VN은 Campus 및 Univ라는 두 개의 VNI가 있는 이전 토폴로지를 기반으로 합니다.

## 변수 정의

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

```
Global_prefixes = 10.10.10.8/30
```

```
FUSION_BGP_AS = 65004
```

```
BORDER_BGP_AS = 65005
```

### VN1의 경우

```
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
```

### VN2의 경우

```
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
```

## 템플릿 예

### 퓨전 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
!

```

## 퓨전 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
```

이 번역에 관하여

Cisco는 전 세계 사용자에게 다양한 언어로 지원 콘텐츠를 제공하기 위해 기계 번역 기술과 수작업 번역을 병행하여 이 문서를 번역했습니다. 아무리 품질이 높은 기계 번역이라도 전문 번역가의 번역 결과물만큼 정확하지는 않습니다. Cisco Systems, Inc.는 이 같은 번역에 대해 어떠한 책임도 지지 않으며 항상 원본 영문 문서(링크 제공됨)를 참조할 것을 권장합니다.