

Catalyst 8000 라우터에서 이더넷 가상 연결 구성

목차

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[배경 정보](#)

[구성](#)

[네트워크 다이어그램](#)

[설정](#)

[예 1. 동일한 VLAN의 호스트 간 패킷 브리지](#)

[예 2. BDI가 호스트 10 및 호스트 20 모두에 대한 기본 게이트웨이 역할을 하도록 구성](#)

[예 3. BDI 인터페이스를 통한 라우팅 구성](#)

[예 4. 재작성 옵션](#)

[다음을 확인합니다.](#)

소개

이 문서에서는 Catalyst 8000 Enterprise Platform에서 EVC(Ethernet Virtual Connections)를 구성하는 실무 예제에 대해 설명합니다.

사전 요구 사항

요구 사항

이 문서에 대한 특정 요건이 없습니다.

사용되는 구성 요소

이 문서의 정보는 Cisco IOS® XE 소프트웨어 버전 17을 기반으로 합니다.

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.

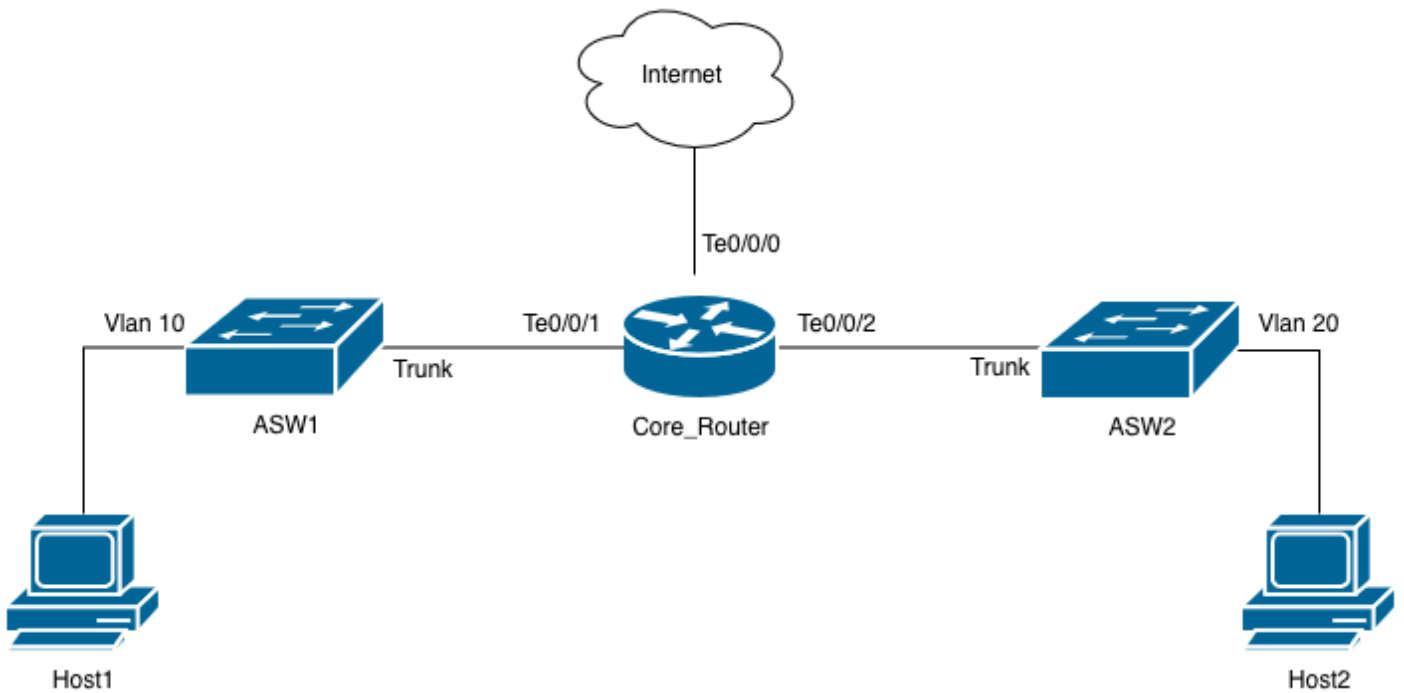
배경 정보

Ethernet Virtual Connections는 사용자 네트워크 인터페이스 간의 엔드 투 엔드 논리적 경로를 표시하여 네트워크 전반에 레이어 2 서비스를 제공하기 위한 유연한 표준 기반 프레임워크를 제공합니다.

이 아키텍처의 핵심은 서비스 인스턴스이며, 이는 VLAN 태그와 같은 특정 기준에 따라 수신 트래픽을 분류하고 이를 적절한 네트워크 서비스에 매핑하는 물리적 인터페이스의 논리적 엔티티로 작동합니다. 이러한 서비스 인스턴스는 브리지 도메인과 연결되며, 이는 이러한 인스턴스 간의 레이어 2 스위칭 또는 라우팅을 용이하게 하는 논리적 브로드캐스트 도메인으로서 서비스 정의를 기본 물리적 인프라와 효과적으로 분리하고 가상 환경 내에서 트래픽 전달 및 정책 애플리케이션에 대한 세분화된 제어를 가능하게 합니다.

구성

네트워크 다이어그램



토폴로지 다이어그램

설정

토폴로지 다이어그램을 고려하십시오. 네트워크의 모든 호스트에 대한 기본 게이트웨이로

Core_Router를 사용하려는 경우 Host1과 Host2는 동일한 VLAN에 있지만 서로 다른 VLAN에 있을 수도 있습니다.

Core_Router에서 하위 인터페이스를 사용하는 경우 두 가지 주요 문제가 있습니다.

- 동일한 VLAN에 속하는 호스트에 대해 서로 다른 서브넷을 구성해야 합니다. 그러면 두 개의 서로 다른 기본 게이트웨이를 사용해야 합니다.
- 각 하위 인터페이스가 브로드캐스트 도메인을 중단하기 때문에 동일한 VLAN 내의 호스트는 호스트 간에 직접 통신할 수 없습니다.

라우팅할 필요가 없는 패킷을 연결하도록 Core_Router를 구성할 수 있으며, 두 호스트의 기본 게이트웨이로 작동하도록 단일 인터페이스를 구성할 수도 있습니다. 이 작업은 EVC(Ethernet Virtual Connections)를 사용하여 수행한 다음 브리지 도메인에 매핑할 수 있습니다. 기본 게이트웨이 역할을 하는 인터페이스를 BDI(Bridge Domain Interface)라고 합니다.

이 예에서는 액세스 스위치가 해당 액세스 VLAN 및 이를 허용하는 트렁크로 이미 구성되어 있습니다.

예 1. 동일한 VLAN에 있는 호스트 간에 패킷을 연결합니다.

```
Core_Router#configure terminal
Core_Router(config)#interface TenGigabitEthernet 0/0/1
Core_Router(config-if)#service instance 10 ethernet
Core_Router(config-if-srv)#encapsulation dot1q 10
Core_Router(config-if-srv)#bridge-domain 10
Core_Router(config-if-srv)#exit
Core_Router(config-if)#exit
Core_Router(config)#
Core_Router(config)#interface TenGigabitEthernet 0/0/2
Core_Router(config-if)#service instance 10 ethernet
Core_Router(config-if-srv)#encapsulation dot1q 10
Core_Router(config-if-srv)#bridge-domain 10
Core_Router(config-if-srv)#exit
Core_Router(config-if)#exit
Core_Router(config)#
Core_Router(config)#bridge-domain 10
Core_Router(config)#end
Core_Router#
```



참고: 서비스 인스턴스 ID는 encapsulation dot1q <vlan-id> 명령으로 지정된 VLAN ID와 같을 필요가 없습니다. 식별 및 문제 해결을 위해 이들을 매치하는 것이 모범 사례입니다.

이제 Host1과 Host2가 서로 ping할 수 있습니다.

예 2. BDI가 호스트 10 및 호스트 20 모두에 대한 기본 게이트웨이 역할을 하도록 구성

<#root>

```
Core_Router#configure terminal
Core_Router(config)#interface bdi 10
Core_Router(config-if)#
```

```
encapsulation dot1q 10
```

```
Core_Router(config-if)#ip address 10.10.10.1 255.255.255.0
Core_Router(config-if)#no shutdown
Core_Router(config-if)#end
Core_Router#
```

이제 BDI에서 두 호스트를 ping할 수 있습니다.

```
Core_Router#ping 10.10.10.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.10.10, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

```
Core_Router#ping 10.10.10.20
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.10.20, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
```

```
Core_Router#show ip arp
Protocol Address Age (min) Hardware Addr Type Interface
Internet 10.10.10.1 - 001e.e525.2fbf ARPA BDI10
Internet 10.10.10.10 0 aabb.cc00.0f00 ARPA BDI10
Internet 10.10.10.20 0 aabb.cc00.1000 ARPA BDI10
```



경고: BDI에서 dot1q 캡슐화를 구성해야 합니다. 기본적으로 브리지 도메인 인터페이스는 VLAN ID가 있는 패킷에 태그를 지정하지 않습니다. 이 컨피그레이션이 적용되지 않을 경우, 패킷은 L2 스위치 트렁크에 도달할 때 네이티브 vlan에 배치됩니다.

예 3. BDI 인터페이스를 통한 라우팅 구성

<#root>

```
Core_Router#configure terminal
```

```
Core_Router(config)#interface bdi 20
Core_Router(config-if)#
```

```
encapsulation dot1q 20
```

```
Core_Router(config-if)# ip address 10.20.20.1 255.255.255.0
Core_Router(config-if)#exit
Core_Router(config)#
Core_Router(config)#
Core_Router(config)#interface TenGigabitEthernet 0/0/2
Core_Router(config-if)#service instance 20 ethernet
Core_Router(config-if-srv)#
```

```
encapsulation dot1q 20
```

```
Core_Router(config-if-srv)#bridge-domain 20
Core_Router(config-if-srv)#exit
Core_Router(config-if)#exit
Core_Router(config)#bridge-domain 20
Core_Router(config)#exit
Core_Router#
```

이제 BDI 20에서 호스트 1을 ping할 수 있습니다.

```
<#root>
```

```
Core_Router#
```

```
ping 10.10.10.10 source bdi 20
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 10.10.10.10, timeout is 2 seconds:
Packet sent with a source address of 10.20.20.1
```

```
!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

```
Core_Router#
```

예 4. 재작성 옵션

dot1q 캡슐화 없이 브리지 도메인 인터페이스를 구성할 수 있습니다. 그러나 L2 스위치가 해당 VLAN ID로 태그가 지정된 패킷을 수신하는지 확인해야 합니다. 마찬가지로, 수신된 패킷이 VLAN 태그 없이 BDI에 전달되는지 확인해야 합니다.

이것이 컨피그레이션의 모습입니다.

<#root>

```
interface TenGigabitEthernet 0/0/1
no ip address
negotiation auto
service instance 10 ethernet
encapsulation dot1q 10
```

```
rewrite ingress tag pop 1 symmetric
```

```
bridge-domain 10
!
```

```
interface TenGigabitEthernet 0/0/2
no ip address
negotiation auto
service instance 20 ethernet
encapsulation dot1q 20
```

```
rewrite ingress tag pop 1 symmetric
```

```
bridge-domain 20
!
```

재작성 방법은 다음과 같습니다.

1. 패킷은 VLAN 태그가 있는 L2 스위치에서 수신됩니다.
2. 이 명령을 사용하면 라우터가 가장 바깥쪽 인그레스 VLAN 태그를 팝업할 수 있습니다.
3. 패킷은 해당 브리지 도메인에 배치됩니다.
4. BDI가 태그 없이 패킷을 수신합니다.
5. 발신 패킷에 대해 반대 작업이 수행됩니다. 이는 마지막에 symmetric 키워드를 구성했기 때문입니다. 이렇게 하면 패킷이 이그레스 시 푸시되는 VLAN 태그를 가져오고 L2 스위치가 태그된 패킷을 받게 됩니다.

다음을 확인합니다.

컨피그레이션이 올바른지 확인하려면 서비스 인스턴스가 구성된 물리적 인터페이스 컨피그레이션과 BDI를 확인합니다.

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.