VMware NSX Segment

세그먼트와 트랜스포트 존

Transport Zone (전송 영역)

Transport Zone는 네트워크 상에서 특정 트래픽이 이동할 수 있는 범위나 "구역"을 정의하는 개념입니다. 쉽게 말해, 큰 도시에서 특정 활동이 이루어지는 "구역"이라고 생각할 수 있습니다. VMware NSX에서는 **Transport Zone**이 특정 호스트(예: ESXi 호스트)들이 어떤 네트워크에 접근할 수 있는지를 정의하는 역할을 합니다.

비유를 들자면, 도시는 여러 구역으로 나뉘어져 있는데, 각각의 구역은 주거, 상업, 산업 등 서로 다른 용도로 사용됩니다. 이 구역들 안에서는 특정 규칙이 적용되고, 필요할 때에는 도로로 연결하여 교통을 할 수 있습니다. **Transport Zone**도 이와 비슷하게 가상 네트워크 상에서 트래픽이 이동할 수 있는 범위를 정해줍니다. 각 구역 안에 있는 네트워크 세그먼트들만 서로 통신할 수 있으며, 구역 간의 통신은 명시적으로 허용되지 않는 한 제한됩니다.

핵심 내용:

- **통신 범위**: 동일한 Transport Zone 안에 있는 호스트들만 해당 Zone 내의 세그먼트들과 통신할 수 있습니다.
- **트래픽 분리**: 다양한 트래픽(예: 개발, 테스트, 운영 트래픽)을 구분하는 데 사용됩니다.
- **연결 제어**: 특정 호스트들이 어떤 네트워크에 접근할 수 있을지 결정하는 데 중요합니다.

Segment (세그먼트)

Segment는 VMware NSX에서 **가상 스위치**나 **가상 네트워크 케이블**이라고 생각하면 됩니다. 이는 가상 머신(VM)이나 워크로드들을 연결하는 논리적인 네트워크입니다. Transport Zone이 하나의 구역이라면, **Segment**는 그 구역 안의 개별 도로에 해당한다고 할 수 있습니다. 이 도로를 통해 데이터가 가상 머신들 사이를 오갈 수 있습니다.

전통적인 네트워크에서는 VLAN(가상 LAN)을 사용하여 물리적인 네트워크 장치들을 그룹화하는데, NSX에서는 **Segment**가 이 역할을 가상화된 형태로 수행합니다. 가상 머신을 서로 연결하거나 다른 네트워크 요소와 연결하는 데 사용됩니다.

예시:

개발 환경에서 애플리케이션을 실행하는 여러 VM이 있을 때, 이들 VM을 같은 **Segment**에 연결하여 서로 통신할 수 있게 할 수 있습니다. 이 **Segment**는 개발 트래픽을 위한

Transport Zone 내에 존재하게 됩니다. 다른 Transport Zone(예: 운영 환경을 위한 Zone)과는 명시적으로 설정하지 않는 한 통신할 수 없습니다.

핵심 내용:

- **가상 네트워크**: **Segment**는 가상 머신을 연결하는 논리적 네트워크로, VLAN과 유사한 역할을 합니다.
- **격리**: Segment는 특정 애플리케이션이나 워크로드를 격리할 수 있습니다.
- **동적 연결**: 네트워크 요구에 따라 Segment를 동적으로 생성, 삭제 또는 수정할 수 있습니다.

Transport Zone과 Segment의 관계

Transport Zone은 대규모 네트워크 상에서 트래픽이 이동할 수 있는 범위를 정의하는 것이고, **Segment**는 그 범위 안에서 가상 머신이나 장치들이 연결되는 세부 네트워크를 정의한다고 볼 수 있습니다. 같은 Transport Zone 안에 있는 세그먼트들만 서로 통신할 수 있으며, 다른 Zone과의 연결은 별도의 설정이 필요합니다.

오버레이(Overlay)란?

- *오버레이(Overlay)**는 물리적인 네트워크 위에 **가상의 네트워크 계층**을 만드는 기술입니다. 이 가상의 계층은 실제 물리적인 네트워크와는 독립적으로 동작하면서, 물리적 네트워크의 제약 없이 더 유연하고 확장 가능한 네트워크 구성을 가능하게 합니다. 간단히 말해, **오버레이 네트워크**는 기존 네트워크 위에 추가로 쌓이는 또 다른 네트워크라고 할 수 있습니다.

예시:

이해를 돕기 위해 비유를 들어보겠습니다. 만약 여러분이 기존의 도로망(물리적 네트워크)을 따라 특정 목적지로 가야 한다고 가정해봅시다. 하지만 그 도로망이 복잡하거나 교통이 혼잡할 수 있죠. 이때, 오버레이 네트워크는 마치 공중에 새로운 도로를 만드는 것과 같습니다. 이 도로는 기존 도로 위에 있지만, 기존의 교통체증이나 구조적인 제한에 영향을 받지 않고, 더 효율적으로 원하는 목적지로 이동할 수 있게 해줍니다.

기술적 설명:

네트워크에서의 오버레이는 주로 **터널링(tunneling)** 프로토콜을 사용하여 구현됩니다. 터널링은 데이터를 물리적인 네트워크 위에서 이동시키는 동시에, 가상 네트워크의 논리적인 경로를 따라 전달되도록 해줍니다. 여기서 주요 터널링 프로토콜 중 하나가 **Geneve**, **VXLAN** 또는 **GRE**입니다. 이러한 프로토콜들은 가상의 네트워크 패킷을 물리적 네트워크 패킷 안에 캡슐화(encapsulation)하여 전송합니다.

오버레이 네트워크의 장점

1. **확장성**: 물리적 네트워크의 IP 주소나 VLAN 같은 제약 없이, 수천 개의 가상 네트워크(세그먼트)를 만들 수 있습니다. 전통적인 VLAN에서는 4096개의 VLAN 제한이 있지만, 오버레이 네트워크는 이보다 훨씬 많은 네트워크를 지원할 수 있습니다.
2. **유연성**: 물리적 네트워크의 물리적인 구조에 의존하지 않고, 가상 네트워크를 쉽게 만들고 수정할 수 있습니다. 예를 들어, 두 데이터센터가 물리적으로 다른 위치에 있어도, 오버레이를 통해 마치 같은 네트워크에 있는 것처럼 가상 머신들이 서로 통신할 수 있습니다.
3. **격리성**: 오버레이 네트워크는 서로 독립적으로 동작하며, 각 가상 네트워크는 다른 네트워크와 격리됩니다. 즉, 여러 고객이나 부서가 같은 물리적인 네트워크를 사용하더라도, 각자의 오버레이 네트워크에서 보안 및 트래픽 관리를 할 수 있습니다.

오버레이 네트워크의 동작 방식

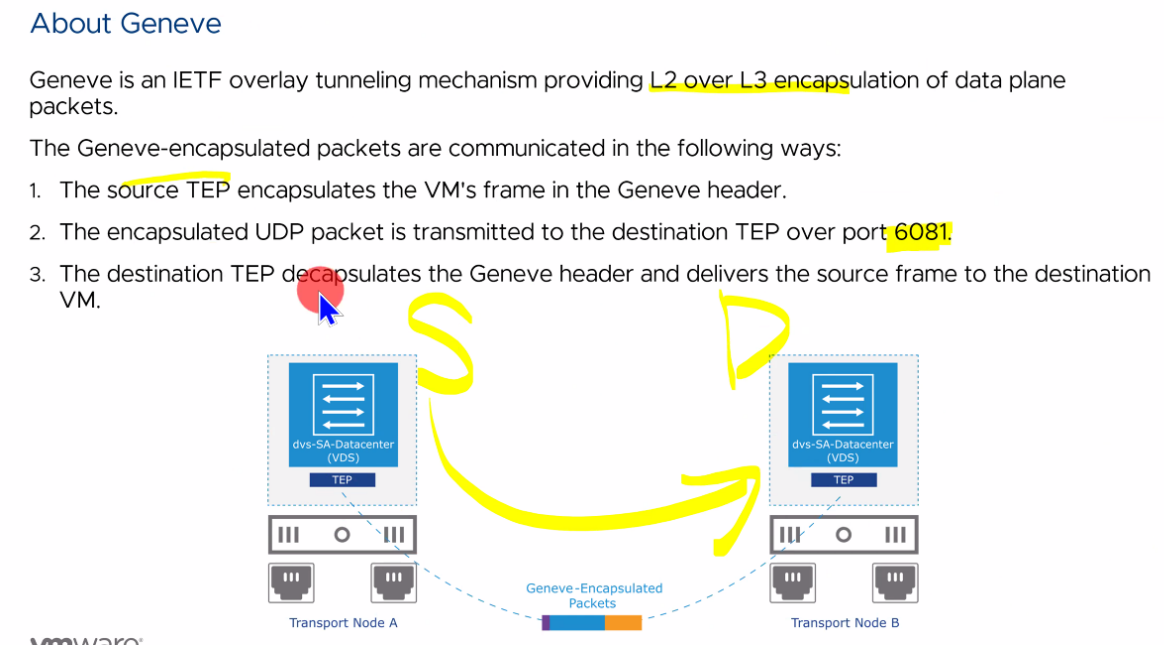
오버레이 네트워크에서 데이터는 **캡슐화(encapsulation)** 과정을 거칩니다. 즉, 가상의 네트워크 패킷(예: 가상 머신 간의 통신 패킷)은 물리적인 네트워크의 패킷에 담겨서 전달됩니다. 이때, 각 가상 네트워크의 특성에 맞는 정보가 패킷 안에 담기고, 물리 네트워크 상에서는 그저 데이터의 흐름으로 처리됩니다.

- **가상 네트워크 패킷**이 기존의 물리 네트워크 패킷 안에 들어갑니다.
- 물리 네트워크는 그저 이 패킷을 목적지로 전달하는 역할만 합니다.
- 목적지에서는 다시 물리 네트워크 패킷을 열어 가상 네트워크 패킷을 추출하고, 이를 처리합니다.

예시:

한 회사가 두 개의 물리적인 데이터센터를 운영한다고 가정해봅시다. 각각의 데이터센터는 서로 멀리 떨어져 있지만, 오버레이 네트워크를 사용하면 두 데이터센터의 가상 머신들이 마치 같은 데이터센터 안에 있는 것처럼 서로 통신할 수 있습니다. 물리적으로는 거리가 있지만, 가상의 터널을 통해 이 둘이 연결되며, 물리적인 경로가 어떻게 생겼는지는 가상 네트워크에 영향을 주지 않습니다.

요약하자면, **오버레이 네트워크**는 기존 물리적 네트워크 위에 논리적으로 쌓여서, 더 유연하고 확장 가능한 네트워크 환경을 제공하는 기술입니다. 물리적 네트워크의 제약에서 벗어나고자 하는 클라우드 데이터센터나 가상 환경에서 필수적인 요소로 사용됩니다.



Geneve 프로토콜이란?

Geneve (Generic Network Virtualization Encapsulation) 프로토콜은 가상 네트워크에서 트래픽을 물리적 네트워크를 통해 캡슐화하여 전달하는 데 사용되는 터널링 프로토콜입니다. 이는 기존의 **VXLAN**이나 **NVGRE**와 같은 다른 터널링 프로토콜들의 장점을 통합하면서, 더 유연하고 확장성 있는 네트워크 가상화 환경을 제공하도록 설계되었습니다.

간단한 비유:

Geneve를 이해하기 위해 쉽게 설명하자면, 데이터를 특정 위치로 보내는 택배 시스템이라고 생각해 볼 수 있습니다. 만약 보내는 물건이 여러 개의 작은 상자로 나눠져 있다면, 이를 하나의 큰 상자에 포장하여 안전하게 목적지로 보낼 수 있습니다. 이 큰 상자가 바로 Geneve 프로토콜을 통해 캡슐화된 데이터이고, 그 내부에 있는 각각의 작은 상자들이 실제 네트워크 패킷입니다. 물리적 네트워크는 이 큰 상자를 목적지까지 안전하게 운반하고, 그 안에 있는 패킷들은 가상 네트워크에서 처리됩니다.

Geneve의 주요 특징

1. **유연한 헤더 구조:**

- **Geneve**는 매우 유연한 헤더 구조를 가지고 있습니다. VXLAN이나 NVGRE와 달리 고정된 필드가 적고, 확장 필드를 자유롭게 추가할 수 있습니다. 이를 통해 각 가상 네트워크에 필요한 맞춤형 정보를 더 쉽게 포함할 수 있습니다.
- 이를 비유하면, 택배 상자에 필요한 만큼 공간을 쉽게 추가하거나 줄일 수 있는 맞춤형 상자라고 생각할 수 있습니다.

2. **다양한 네트워크 유형 지원**:
   - **Geneve**는 다양한 유형의 네트워크를 지원합니다. 데이터센터 내에서 사용하는 L2(Layer 2) 네트워크, L3(Layer 3) 네트워크뿐만 아니라 멀티캐스트나 유니캐스트 등 다양한 통신 방식을 지원할 수 있습니다.
   - 이 점에서 Geneve는 상황에 맞게 유연하게 변형할 수 있는 전천후 네트워크 연결 방식이라 할 수 있습니다.
3. **캡슐화(Encapsulation)**:
   - Geneve는 **캡슐화 프로토콜**로, 기존의 물리 네트워크 위에 가상의 네트워크 트래픽을 전달합니다. 가상의 네트워크 패킷을 물리적인 네트워크 패킷 안에 넣어 다른 물리적 호스트로 안전하게 전달하는 역할을 합니다.
   - 이때 캡슐화는 물리적인 네트워크의 특성이나 제약에 상관없이, 가상의 네트워크가 원활히 동작할 수 있게 하는 중요한 역할을 합니다.
4. **확장성**:
   - Geneve는 대규모 클라우드 환경에서 수천 개의 가상 네트워크를 처리할 수 있도록 설계되었습니다. 이로 인해 데이터센터 또는 대규모 클라우드 서비스 제공자들이 사용하는 중요한 기술입니다.
   - 마치 하나의 도로에서 수많은 차들이 목적지로 빠르게 이동할 수 있도록 돕는 교통 시스템과 같다고 생각할 수 있습니다.
5. **다양한 기능 통합**:
   - Geneve는 다른 터널링 프로토콜들의 장점을 통합합니다. VXLAN의 확장성, NVGRE의 효율적인 멀티캐스트 지원, STT(Stream Control Transmission Protocol)의 성능 이점을 모두 수용하며, 다양한 사용 사례에 대응할 수 있습니다.

Geneve 패킷 구조

Geneve 패킷은 물리 네트워크 패킷에 담겨 목적지로 전달됩니다. 이 패킷은 여러 개의 헤더를 포함하며, 이는 다음과 같이 구성됩니다:

1. **Outer Ethernet Header**: 물리적 네트워크에서의 목적지를 정의합니다.

2. **Outer IP Header**: 패킷이 물리적인 IP 네트워크를 통해 이동할 때 사용되는 IP 주소를 포함합니다.
3. **UDP Header**: Geneve가 UDP(사용자 데이터그램 프로토콜)를 기반으로 하므로, 이 프로토콜을 사용하여 빠른 전송을 가능하게 합니다.
4. **Geneve Header**: 가장 중요한 부분으로, 가상 네트워크의 메타데이터와 확장 필드를 포함합니다. 이 헤더가 네트워크 가상화의 여러 특성을 정의하고 확장성을 제공합니다.
5. **Inner Payload**: 실제로 전달해야 하는 가상 네트워크 트래픽(즉, 가상 머신 간의 통신 데이터)이 이 부분에 들어갑니다.

이 구조는 매우 유연하며, 필요에 따라 더 많은 메타데이터를 추가하거나 확장할 수 있습니다.

Geneve의 장점

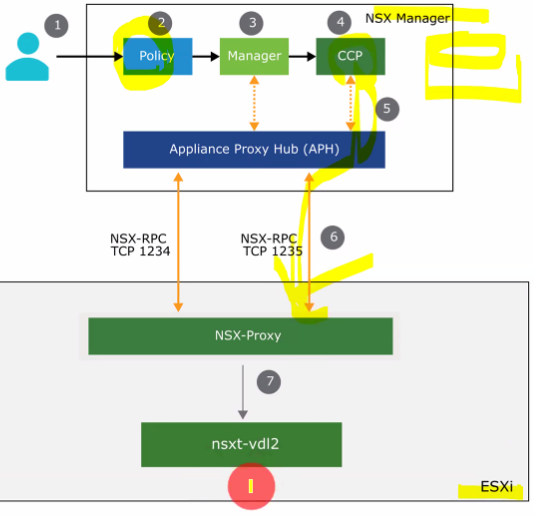
1. **높은 유연성**: 다른 터널링 프로토콜들에 비해 Geneve는 다양한 확장 옵션을 제공하여 새로운 기능이나 요구사항을 쉽게 통합할 수 있습니다.
2. **확장성**: 대규모 클라우드 환경에서 네트워크를 확장하는 데 최적화되어 있습니다.
3. **유연한 캡슐화**: 물리적 네트워크 상에서 가상 네트워크를 효율적으로 캡슐화하여 전달할 수 있습니다.
4. **표준화된 프로토콜**: Geneve는 IETF(Internet Engineering Task Force)에서 표준화된 프로토콜로, 다양한 벤더에서 널리 사용되고 있습니다.

Geneve 프로토콜은 기본적으로 **UDP 6081번 포트**를 사용합니다.

Creating Segments Workflow

To create segments:

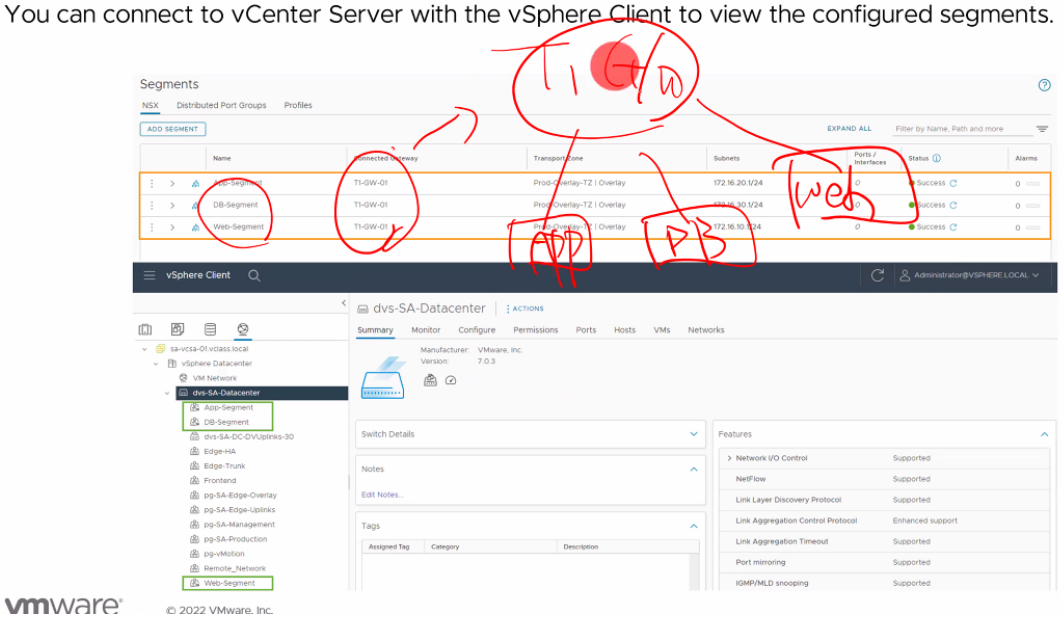
1. A user creates a segment through the NSX UI.
2. The policy role pushes the configuration to the manager role.
3. NSX Manager realizes the segment information as logical switches in the Corfu database.
4. The manager role forwards the segment information to the CCP.
5. The CCP sends the information to the APH.
6. The APH service sends the switching configuration to the local control plane (nsx-proxy) over port 1235.
7. The nsx-proxy agent forwards the switching configuration to the nsxt-vdl2 kernel module, which creates and configures the segments in the datapath.



- 세그먼트 워크플로우

Viewing Configured Segments

You can connect to vCenter Server with the vSphere Client to view the configured segments.

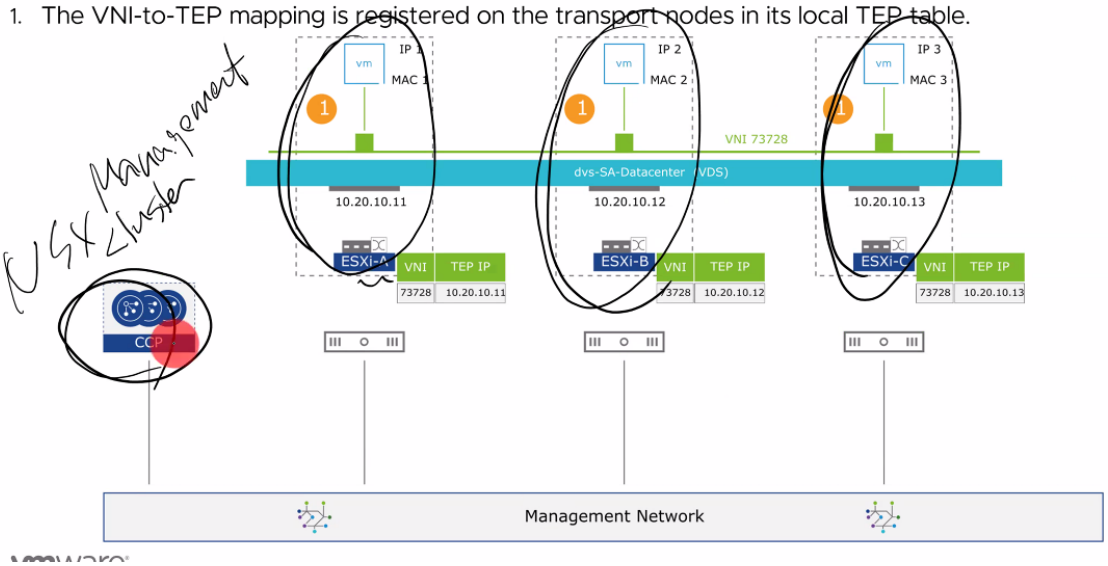


- 티어1 게이트웨이로 각각의 VM을 가진 호스트에 세그먼트가 하나씩 생김

TEP Table Update (1)

When a powered-on VM is connected to a segment:

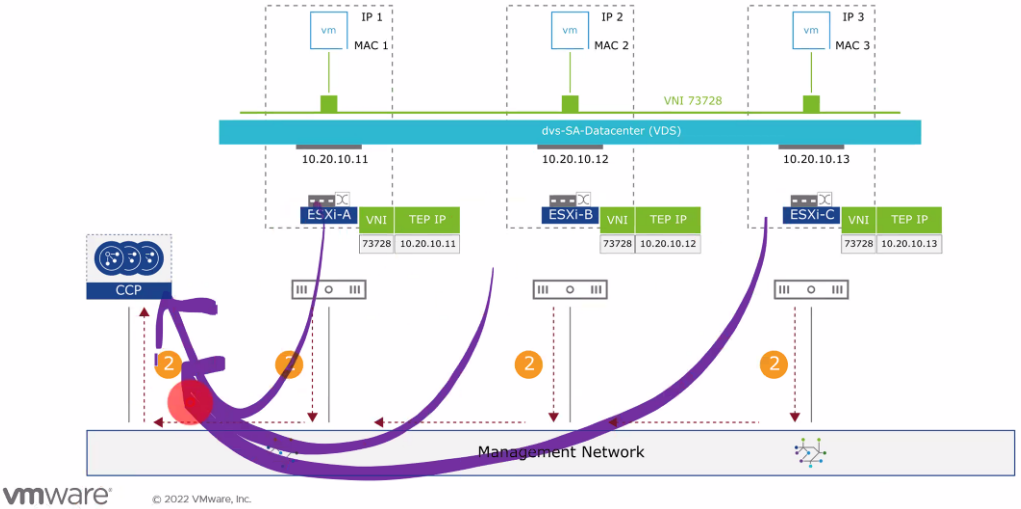
1. The VNI-to-TEP mapping is registered on the transport nodes in its local TEP table.



- management 네트웍으로 각 호스트가 연결되어 있음
- CCP는 매니지먼트 클러스터 매니저
- VNI 세그먼트가 가지고 있는 number
- VNI - TEP IP 매핑
- TEP 테이블이 VNI 터널용 세그먼트 넘버를 매핑해서 로우로 가지고 있음

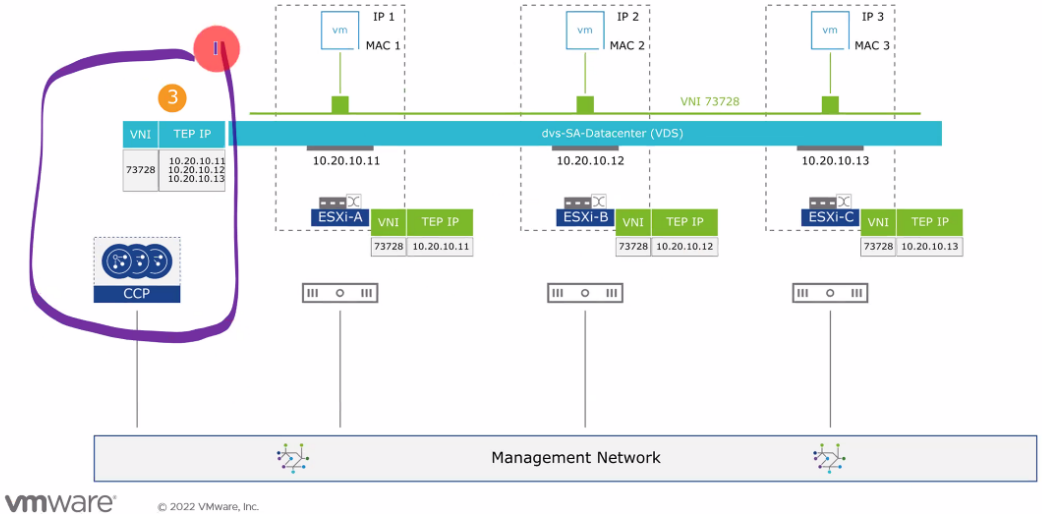
TEP Table Update (2)

2. Each transport node updates the CCP about the learned VNI-to-TEP IP mapping.



TEP Table Update (3)

3. The CCP maintains the consolidated entries of VNI-to-TEP IP mappings.

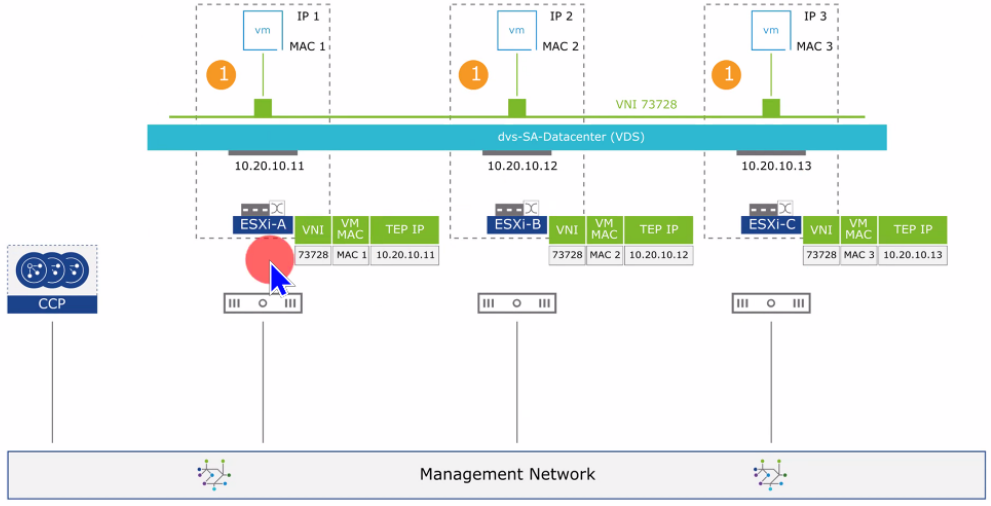


- 각 호스트들은 VNI, TEP IP를 매핑해서 CCP로 뿌려준다.
- 그리고 CCP도 호스트로 다시 뿌려주어 오버라이트 되며 싱크를 맞춘다.

MAC Table Update (1)

When a powered-on VM is connected to a segment:

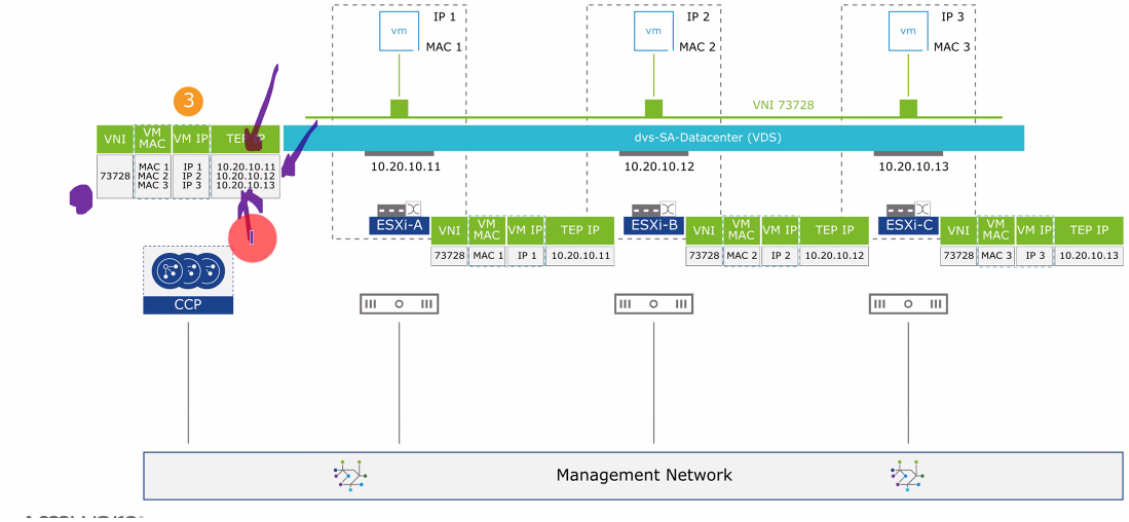
1. The VM MAC-to-TEP IP mapping is registered on the transport nodes in its local MAC table.



- 맥도 뿌리고 업데이트한다.

ARP Table Update (3)

3. The CCP updates its ARP table based on the VM IP-to-MAC mappings received from transport nodes.



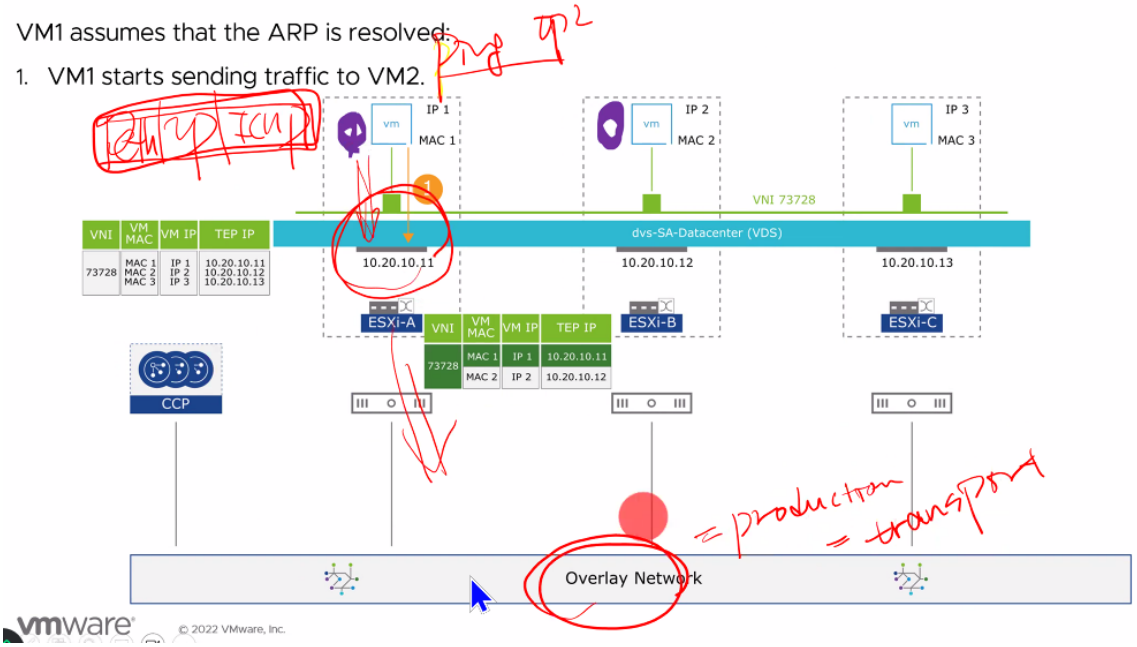
- 최종적으로 CCP는 VNI, VM IP, MAC, TEP IP 등을 가지고 있다.
- 호스트는 LCP에 Table 보관

ARP를 안 뿌리는 이유

Unicast Packet Forwarding Across Hosts (1)

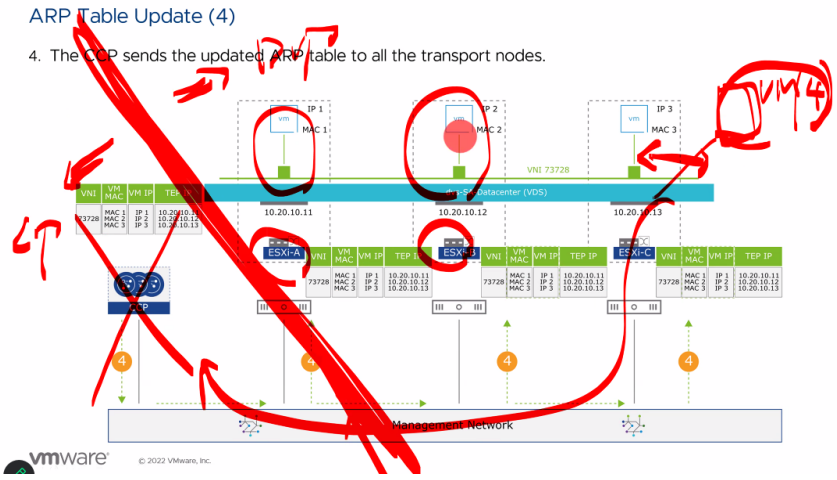
VM1 assumes that the ARP is resolved.

1. VM1 starts sending traffic to VM2.



- CCP에게 받은 TAP 테이블정보로 mac, IP, 등을 이미 알고 있다.
- 실제로 데이터를 주고 받을 때 오버레이를 사용한다.
- CCP는 오버레이랑 붙이지 않는다.
- ARP는 컨트롤 플레인이고
- 유니캐스트는 데이터 플레인 동작이다.
- CCP가 죽어도 LCP가 있기에 호스트간 통신 가능

SDN 솔루션의 특징



- 컨트롤 플레인과 데이터 플레인을 분리한다.
- SDN 솔루션 사용시 네트워크 이점

SDN (Software-Defined Networking) 솔루션 사용 시 네트워크의 이점

- *SDN (Software-Defined Networking)**은 네트워크의 **제어 플레인**과 **데이터 플레인**을 분리하여, 네트워크 인프라를 중앙에서 소프트웨어적으로 관리하고 제어할 수 있는 기술입니다. SDN 솔루션을 사용하면 물리적인 하드웨어 장치의 제약에서 벗어나 보다 유연하고 효율적으로 네트워크를 운영할 수 있습니다. 이를 통해 다양한 이점이 발생하는데, 이를 하나씩 살펴보겠습니다.

1. 중앙 집중식 관리

SDN을 사용하면 네트워크의 모든 장치와 트래픽을 중앙에서 관리할 수 있습니다. 전통적인 네트워크에서는 스위치, 라우터 등 개별 장치를 하나하나 설정하고 관리해야 하지만, SDN에서는 모든 네트워크 장치를 중앙의 **SDN 컨트롤러**에서 제어합니다.

비유:

이것을 한 도시의 교통 시스템으로 비유하자면, 전통적인 네트워크에서는 각각의 신호등이 독립적으로 운영되지만, SDN에서는 중앙에서 모든 신호등을 한 번에 제어하고, 교통 상황에 따라 실시간으로 최적의 신호 체계를 조정하는 것과 같습니다.

이점:

- 관리 효율성 향상: 하나의 콘솔에서 전체 네트워크를 관리하고 변경 사항을 쉽게 적용할 수 있습니다.
- 빠른 문제 해결: 네트워크 상에서 발생하는 문제를 중앙에서 쉽게 추적하고 수정할 수 있습니다.

---

2. 유연성 및 네트워크 자동화

SDN을 통해 네트워크를 소프트웨어적으로 정의하고 제어할 수 있기 때문에, 새로운 네트워크 정책을 쉽게 설정하고 변경할 수 있습니다. 트래픽 경로를 재설정하거나 새로운 장치를 추가할 때, 물리적 네트워크 장치의 설정을 변경하지 않고 소프트웨어로 처리할 수 있습니다.

비유:

기존의 네트워크 설정이 벽돌로 지어진 고정된 건물이라면, SDN은 레고 블록처럼 언제든지 빠르게 조립하고 해체할 수 있는 구조라고 생각할 수 있습니다.

이점:

- 네트워크 확장성: 네트워크 환경이 확장되거나 축소되더라도 쉽게 대응할 수 있습니다.
- 자동화 가능: 반복적인 설정 작업을 자동화할 수 있어 운영 비용을 절감할 수 있습니다.

---

3. 향상된 트래픽 제어 및 최적화

SDN은 네트워크 트래픽을 실시간으로 모니터링하고, 필요에 따라 트래픽 경로를 동적으로 조정할 수 있습니다. 이를 통해 네트워크 혼잡을 줄이고, 대역폭을 최적화하여 트래픽 흐름을 원활하게 유지할 수 있습니다.

비유:

이것은 마치 스마트 교통 시스템이 교통량을 실시간으로 모니터링하고, 혼잡한 도로를 피하도록 차량을 다른 경로로 안내하는 것과 유사합니다.

이점:

- 대역폭 최적화: 중요한 트래픽에 우선순위를 부여하고 불필요한 트래픽을 제어할 수 있습니다.
- 네트워크 성능 향상: 트래픽 경로를 동적으로 조정함으로써 네트워크 지연 시간을 줄이고 성능을 향상시킬 수 있습니다.

---

4. 보안 강화

SDN은 네트워크의 보안을 강화하는 데 매우 유용합니다. 중앙에서 네트워크 전체를 제어할 수 있기 때문에, 보안 정책을 일관되게 적용하고, 빠르게 변경할 수 있습니다. 또한, SDN은 네트워크 트래픽을 세분화하여, 세그먼트 간에 강력한 격리와 제어를 제공할 수 있습니다.

비유:

전통적인 네트워크 보안이 여러 개의 자물쇠를 사용해 각 방을 보호하는 것이라면, SDN은 보안 카메라와 경보 시스템으로 전체 건물을 실시간으로 모니터링하며 위협을 차단하는 방식입니다.

이점:

- 네트워크 세분화: 민감한 트래픽을 격리하여 보안을 강화할 수 있습니다.
- 실시간 보안 업데이트: 새로운 보안 위협이 발생했을 때, 중앙에서 빠르게 보안 정책을 수정하고 배포할 수 있습니다.

---

5. 비용 절감

SDN은 네트워크 하드웨어에 대한 의존도를 줄여주기 때문에, 더 저렴한 범용 하드웨어로도 고성능 네트워크를 구축할 수 있습니다. 또한, 네트워크 관리 작업을 자동화하고 중앙 집중식으로 관리함으로써 운영 비용도 절감됩니다.

비유:

전통적인 네트워크에서는 고급 맞춤형 장비(비싼 맞춤형 가구)를 사용해야 하지만, SDN에서는 저렴한 범용 장비(기본적인 가구)만으로도 충분히 운영할 수 있습니다. 필요한 경우, 소프트웨어적으로 기능을 추가하거나 변경하면 됩니다.

이점:

- 저렴한 하드웨어 사용 가능: 범용 장비로 네트워크를 구축할 수 있어 초기 비용 절감.

- 운영 비용 절감: 자동화와 중앙 집중화 덕분에 운영 및 유지 보수 비용이 감소합니다.

---

6. 확장성 및 클라우드 환경에서의 최적화

SDN은 클라우드 환경과 완벽하게 호환됩니다. 클라우드 기반 인프라는 빠르게 확장되거나 축소될 수 있어야 하는데, SDN은 이러한 변화에 매우 유연하게 대응할 수 있습니다. 또한, SDN은 멀티테넌시를 쉽게 지원하여 여러 사용자가 각자 독립적인 네트워크를 사용할 수 있도록 도와줍니다.

비유:

클라우드 환경에서의 네트워크를 물리적 네트워크로 운영하는 것은 마치 고정된 책장에 책을 쌓는 것과 비슷합니다. 하지만 SDN은 책장을 계속해서 확장하거나 줄일 수 있는 시스템과 같아, 클라우드의 유동성에 맞춰 쉽게 조정할 수 있습니다.

이점:

- 클라우드 환경 최적화: SDN은 클라우드 서비스의 확장성과 유연성을 최대한 활용할 수 있도록 지원합니다.
- 멀티테넌시 지원: 여러 고객이나 부서가 각각의 네트워크를 독립적으로 사용할 수 있습니다.

요약

SDN 솔루션을 사용하면 다음과 같은 네트워크 이점을 얻을 수 있습니다:

1. **중앙 집중식 관리**로 네트워크의 통제와 문제 해결이 용이.
2. **유연성과 자동화**로 네트워크 설정 및 변경이 신속하게 이루어짐.
3. **트래픽 최적화**를 통해 네트워크 성능을 향상시키고 대역폭을 효율적으로 사용.
4. **보안 강화**로 네트워크 트래픽을 세분화하고, 실시간으로 보안 정책을 적용.
5. **비용 절감**을 통해 초기 장비 비용과 운영 비용을 줄임.
6. **확장성**이 뛰어나 클라우드 및 대규모 네트워크 환경에 적합.

SDN 솔루션을 사용함으로써 기업이나 클라우드 환경에서 네트워크를 효율적이고 유연하게 운영할 수 있으며, 빠르게 변화하는 비즈니스 요구에 적응할 수 있는 유연한 네트워크 인프라를 구축할 수 있습니다.