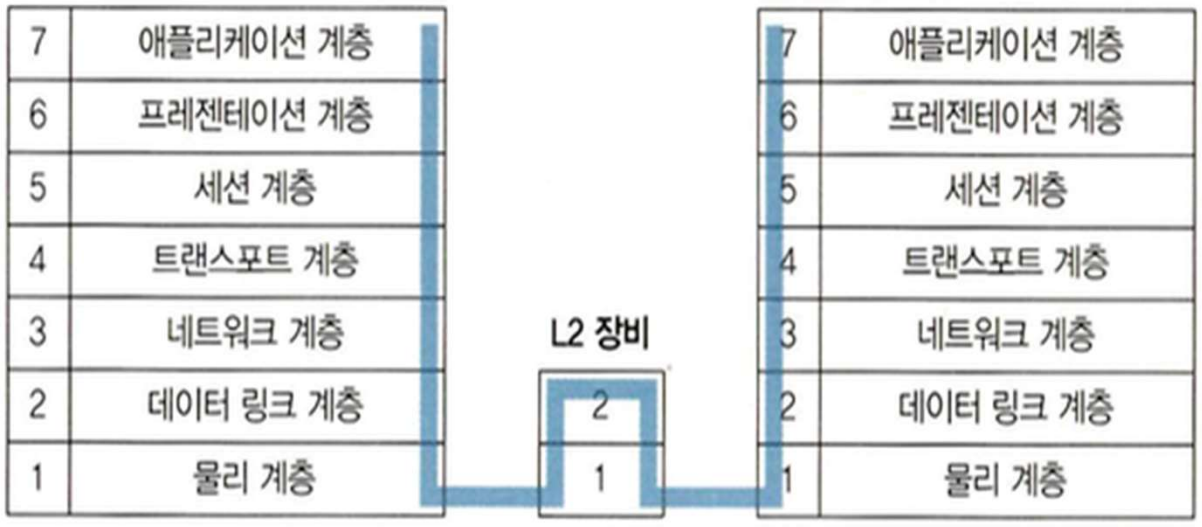
네트워크 장비

Switch

❖ 장비 동작

✓ 개요

- 스위치는 네트워크 중간에서 패킷을 받아 필요한 곳에만 보내주는 네트워크의 중재자 역할을 수행
- 스위치는 아무 설정 없이 네트워크에 연결해도 MAC 주소를 기반으로 패킷을 전달하는 기본 동작을 수행할 수 있음
- 스위치는 MAC 주소를 인식하고 패킷을 전달하는 스위치의 기본 동작 외에도 한 대의 장비에서 논리적으로 네트워크를 분리할 수 있는 VLAN 기능과 네트워크의 루프를 방지하는 스패닝 트리 프로토콜(STP)과 같은 기능을 기본적으로 가지고 있으며 다양한 보안 기능과 모니터링에 필요한 여러 기능이 있음



Switch

❖ 장비 동작

✓ Packet & Frame

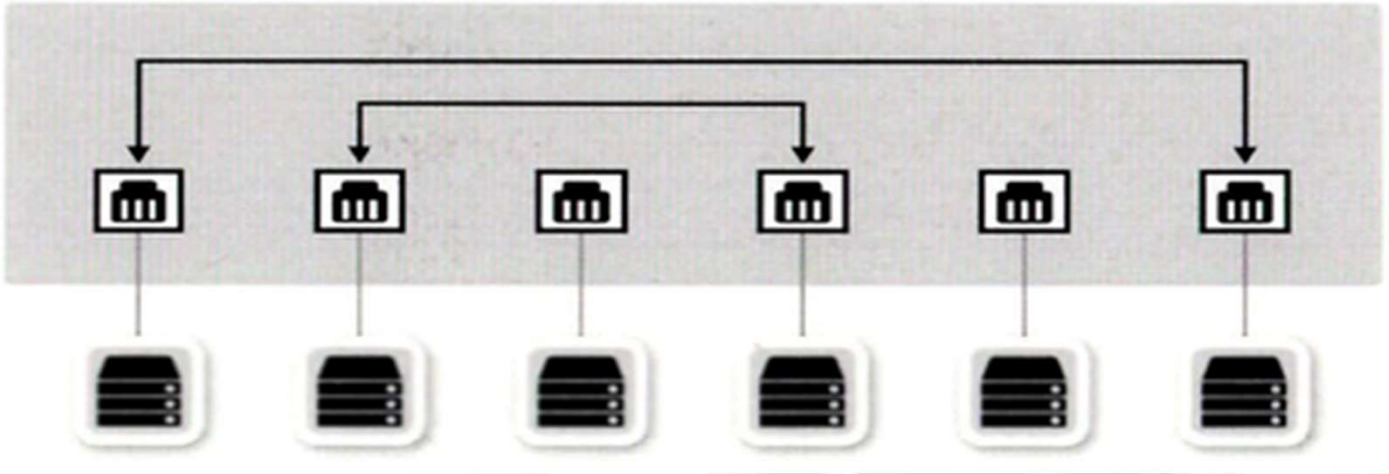
- 각 계층에서 헤더와 데이터를 합친 부분을 PDU(Protocol Data Unit) 라고 부름
- 각 계층 마다 이 PDU를 부르는 이름이 달라서 1계층 PDU는 비트(Bits) 2계층 PDU는 프레임(Frame) 3계층은 패킷(Packet) 4계층은 세그먼트(Segment) 라고 부르고 애플리케이션에 해당하는 3개 계층은 데이터(Data) 라고 부름

계층	데이터(PDU)
애플리케이션 계층	데이터
프레젠테이션 계층	데이터
세션 계층	데이터
트랜스포트 계층	세그먼트
네트워크 계층	패킷
데이터 링크 계층	프레임
물리 계층	비트

* PDU: Protocol Data Unit

Switch

- ❖ 장비 동작
  - ✓ 역할
    - 스위치는 네트워크에서 통신을 중재하는 장비
    - 스위치가 없던 오래된 이더넷 네트워크에서는 패킷을 전송할 때 서로 경합해 그로 인한 네트워크 성능 저하가 컸음
    - 이런 경쟁을 없애고 패킷을 동시에 여러 장비가 서로 간섭 없이 통신하도록 도와주는 장비가 스위치
    - 스위치를 사용하면 여러 단말이 한꺼번에 통신할 수 있어 통신하기 위해 기다리거나 충돌 때문에 대기하는 문제가 해결되고 네트워크 전체의 통신 효율성이 향상됨



Switch

❖ 장비 동작

✓ 역할

- 스위치는 네트워크에서 통신을 중재하는 장비
- 스위치가 없던 오래된 이더넷 네트워크에서는 패킷을 전송할 때 서로 경합해 그로 인한 네트워크 성능 저하가 컸음
- 이런 경쟁을 없애고 패킷을 동시에 여러 장비가 서로 간섭 없이 통신하도록 도와주는 장비가 스위치
- 스위치를 사용하면 여러 단말이 한꺼번에 통신할 수 있어 통신하기 위해 기다리거나 충돌 때문에 대기하는 문제가 해결되고 네트워크 전체의 통신 효율성이 향상됨
- 스위치의 핵심 역할은 누가 어느 위치에 있는지 파악하고 실제 통신이 시작되면 자신이 알고 있는 위치로 패킷을 정확히 전송하는 것으로 이런 동작은 스위치가 2계층 주소를 이해하고 단말의 주소인 MAC 주소와 단말이 위치하는 인터페이스 정보를 매핑한 MAC 주소 테이블을 갖고 있어서 가능
- MAC Address Table

MAC 주소	포트
1111:2222:3333	Eth 1
4444:5555:6666	Eth 2
7777:8888:9999	Eth 3
AAAA:BBBB:CCCC	Eth 4

Switch

- ❖ 장비 동작
  - ✓ 역할
    - 스위치는 전송하려는 패킷의 헤더 안에 있는 2계층 목적지 주소를 확인하고 MAC 주소 테이블에서 해당 주소가 어느 포트에 있는지 확인해 해당 패킷을 그 포트로만 전송하는데이런 역할을 수행하기 위해 스위치는 MAC 주소와 포트가 매핑된 MAC 주소 테이블이 필요
    - 테이블에 없는 도착지 주소를 가진 패킷이 스위치로 들어오면 스위치는 전체 포트로 패킷을 전송
    - 패킷의 도착지 주소가 테이블에 있으면 해당 주소가 매핑된 포트로만 패킷을 전송하고 다른 포트로는 전송하지 않음
    - 스위치의 이런 동작 방식
      - ◆ 플러딩(Flooding)
      - ◆ 어드레스 러닝(Address Learning)
      - ◆ 포워딩/필터링(Forwarding/Filtering)

Switch

- ❖ 장비 동작
  - ✓ Flooding
    - 스위치는 부팅하면 네트워크 관련 정보가 아무 것도 없는데 이때 스위치는 네트워크 통신을 중재하는 자신의 역할을 하지 못하고 허브처럼 동작
    - 허브는 패킷이 들어온 포트를 제외하고 모든 포트로 패킷을 전달
    - 스위치가 허브와 같이 모든 포트로 패킷을 흘리는 동작 방식을 플러딩이라고 함
    - 스위치는 패킷이 들어오면 도착지 MAC 주소를 확인하고 자신이 갖고 있는 MAC 주소 테이블에서 해당 MAC 주소가 있는지 확인
    - MAC 주소 테이블에 매칭되는 목적지 MAC 주소 정보가 없으면 모든 포트에 같은 내용의 패킷을 전송
    - 스위치는 LAN에서 동작하므로 자신이 정보를 갖고 있지 않더라도 어딘가에 장비가 있을 수 있다고 가정하고 이와 같은 작업을 수행
    - 플러딩 동작은 스위치의 정상적인 동작이지만 이런 동작이 많아지면 스위치가 제 역할을 못하게 됨
    - 패킷이 스위치에 들어오면 해당 패킷 정보의 MAC 주소를 보고 이를 학습해 MAC 주소 테이블을 만든 후 이를 통해 패킷을 전송

Switch

- ❖ 장비 동작
  - ✓ Flooding
    - ● 비정상적인 플러딩
      - ◆ 스위치가 패킷을 플러딩한다는 것은 스위치가 제 기능을 못한다는 의미
      - ◆ 이더넷-TCP/IP 네트워크에서는 ARP 브로드캐스트를 미리 주고받은 후 데이터가 전달되므로 실제로 데이터를 보내고 받을 때는 스위치가 패킷을 플러딩하지 않음
      - ◆ 스위치를 사용하면 필요한 곳에만 패킷을 포워딩하므로 주변 통신을 악의적으로 가로채기 힘들어 모든 패킷을 플러딩하는 허브에 비해 보안에 도움이 됨
      - ◆ 이런 스위치 기능을 무력화해 주변 통신을 모니터링하는 공격 기법이 사용되는데 스위치에게 엉뚱한 MAC 주소를 습득시키거나 스위치의 MAC 테이블을 꽉 차게 해 스위치의 플러딩 동작을 유도할 수 있음
      - ◆ 아무 이유없이 스위치가 패킷을 플러딩한다면(주변 다른 통신)을 모니터링할 수 있다면 스위치가 정상적으로 동작하지 않거나 주변에서 공격이 수행되는 상황임을 알아야 함
      - ◆ 이 외에도 ARP 포이즈닝(Poisoning) 기법을 이용해 모니터링해야 할 IP의 MAC 주소가 공격자 자신인 것 처럼 속여 원하는 통신을 받는 방법을 사용하기도 함

Switch

- ❖ 장비 동작
  - ✓ Address Learning
    - 스위치가 패킷의 도착지 MAC 주소를 확인하여 원하는 포트로 포워딩하는 스위치의 동작을 정상적으로 수행하려면 MAC 주소 테이블을 만들고 유지해야 함
    - MAC 주소 테이블은 어느 위치(포트)에 어떤 장비(MAC 주소)가 연결되었는지에 대한 정보가 저장되어 있는 임시 테이블
    - MAC 주소 테이블을 만들고 유지하는 과정이 어드레스 러닝
    - 어드레스 러닝은 패킷의 출발지 MAC 주소 정보를 이용
    - 패킷이 특정 포트에 들어오면 스위치에는 해당 패킷의 출발지 MAC 주소와 포트 번호를 MAC 주소 테이블에 기록
    - 1번 포트에서 들어온 패킷의 출발지 MAC 주소가 AAAA라면 1번 포트에 AAAA MAC 주소를 가진 장비가 연결되어 있다고 추론할 수 있어 이런 방법으로 주소 정보를 습득
    - 어드레스 러닝은 출발지의 MAC 주소 정보를 사용하므로 브로드캐스트나 멀티캐스트에 대한 MAC 주소를 학습할 수 없음
    - 두 가지 모두 목적지 MAC 주소 필드에서만 사용하기 때문

Switch

- ❖ 장비 동작
  - ✓ Address Learning
    - ● 사전 정의된 MAC Address Table
      - ◆ 스위치는 MAC 어드레스 러닝 작업으로 주변 장비의 MAC 주소를 학습하는 것 외에도 사전에 미리 정의된 MAC 주소 정보를 가지고 있음
      - ◆ 사전 정의된 주소는 패킷을 처리하기 위한 주소가 아니라 대부분 스위치 간 통신을 위해 사용되는 주소
      - ◆ 이런 종류의 주소도 MAC 주소 테이블에서 정보를 확인할 수 있는데 스위치에서 자체 처리되는 주소는 특정 포트로 내보내는 것 이 아니라 스위치에서 자체 처리 하므로 인접 포트 정보가 없거나 CPU 혹은 관리 모듈을 지칭하는 용어로 표기됨

Switch

- ❖ 장비 동작
  - ✓ Forwarding/Filtering
    - 패킷이 스위치에 들어온 경우 도착지 MAC 주소를 확인하고 자신이 가진 MAC 테이블과 비교해 맞는 정보가 있으면 매치되는 해당 포트로 패킷을 포워딩하는데 이때 다른 포트로는 해당 패킷을 보내지 않으므로 이 동작이 필터링
    - 스위치는 이런 포워딩과 필터링을 통해 목적지로만 패킷이 전달되도록 동작
    - 스위치에서는 포워딩 과 필터링 작업이 여러 포트에서 동시에 수행될 수 있음
    - 통신이 다른 포트에 영향을 미치지 않으므로 다른 포트에서 기존 통신작업으로 부터 독립 적으로 동작할 수 있음
    - 스위치는 일반적인 유니캐스트에 대해서만 포워딩과 필터링 작업을 수행하며 BUM 트래픽이라고 부르는 브로드캐스트와 언노운 유니캐스트 그리고 멀티캐스트는 조금 다르게 동작
    - 어드레스 러닝 과정에 대해 앞에서 알아보았듯이 출발지 MAC 주소로 브로드캐스트나 멀티캐스트 모두 출발지가 사용되지 않으므로 이런 트래픽은 전달이나 필터링 작업을 하지 않고 모두 플러딩
    - 언노운 유니캐스트도 MAC 주소 테이블에 없는 주소이므로 브로드캐스트와 동일하게 플러딩

Switch

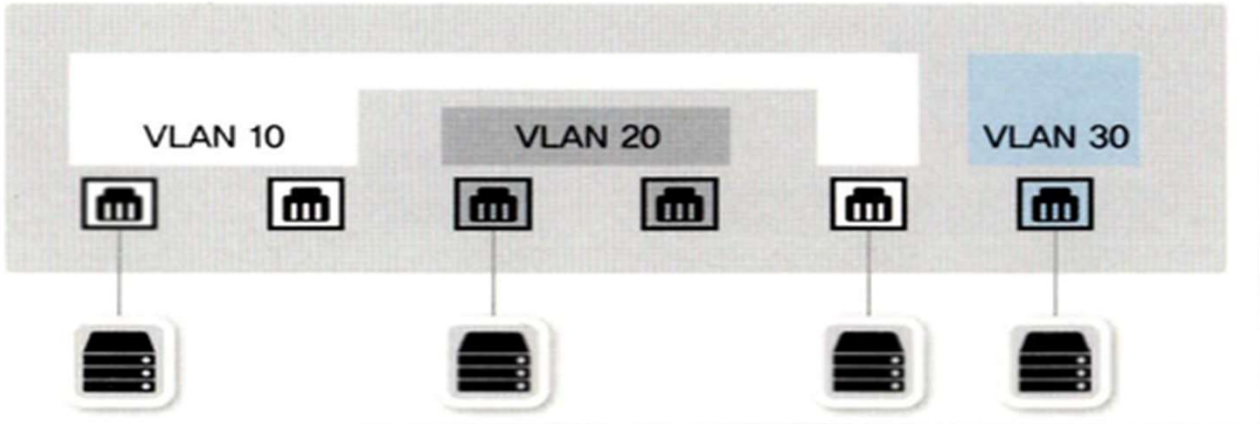
- ❖ 장비 동작
  - ✓ LAN에서의 ARP - 스위치 동작
    - 이더넷 - TCP/IP 네트워크에서는 스위치가 유니캐스트를 플러딩하는 경우는 거의 없음
    - 패킷을 만들기 전에 통신해야 하는 단말의 MAC 주소를 알아내기 위해 ARP 브로드캐스트가 먼저 수행되어야 하므로 유니캐스트보다 ARP 브로드캐스트가 먼저 네트워크에 전달되는데 이 ARP를 이용한 MAC 주소 습득 과정에서 이미 스위치는 통신하는 출발지와 목적지의 MAC 주소를 습득할 수 있고 실제 유니캐스트 통신이 시작되면 이미 만들어진 MAC 주소 테이블로 패킷을 포워딩 및 필터링
    - ARP와 MAC 테이블은 일정 시간 동안 지워지지 않는데 이 시간을 에이징 타임(Aging Time)이라고 하는데 일반적으로 MAC 테이블의 에이징 타임이 단말의 ARP 에이징 타임보다 길어 이더넷 네트워크를 플러딩 없이 효율적으로 운영할 수 있음

Switch

❖ VLAN

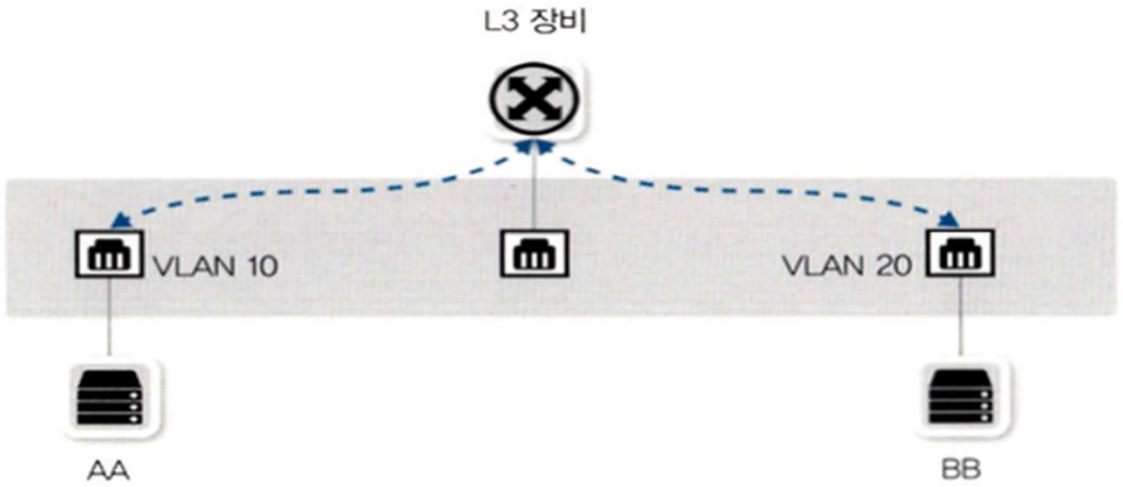
✓ 개요

- VLAN은 물리적 배치와 상관없이 LAN을 논리적으로 분할 구성하는 기술
- 기업과 같이 여러 부서가 함께 근무하면서 각 부서 별로 네트워크를 분할할 때는 네트워크를 여러 개 운영해야 함
- 최근에는 전화기, 복합기, 스마트폰과 같이 PC 외에도 다수의 단말이 네트워크에 연결되므로 네트워크 분할이 더 중요
- 과도한 브로드캐스트로 인한 단말들의 성능 저하, 보안 향상을 위한 차단 용도, 서비스 성격에 따른 정책 적용과 같은 이유로 네트워크가 분리되어야 함
- VLAN을 나누면 하나의 장비를 서로 다른 네트워크를 갖도록 논리적으로 분할한 것이므로 유니캐스트 뿐 아니라 브로드캐스트도 VLAN 간에 통신할 수 없음



Switch

- VLAN
  - ✓ 개요
    - VLAN 간의 통신이 필요하다면 서로 다른 네트워크 간의 통신이므로 3계층 장비가 필요

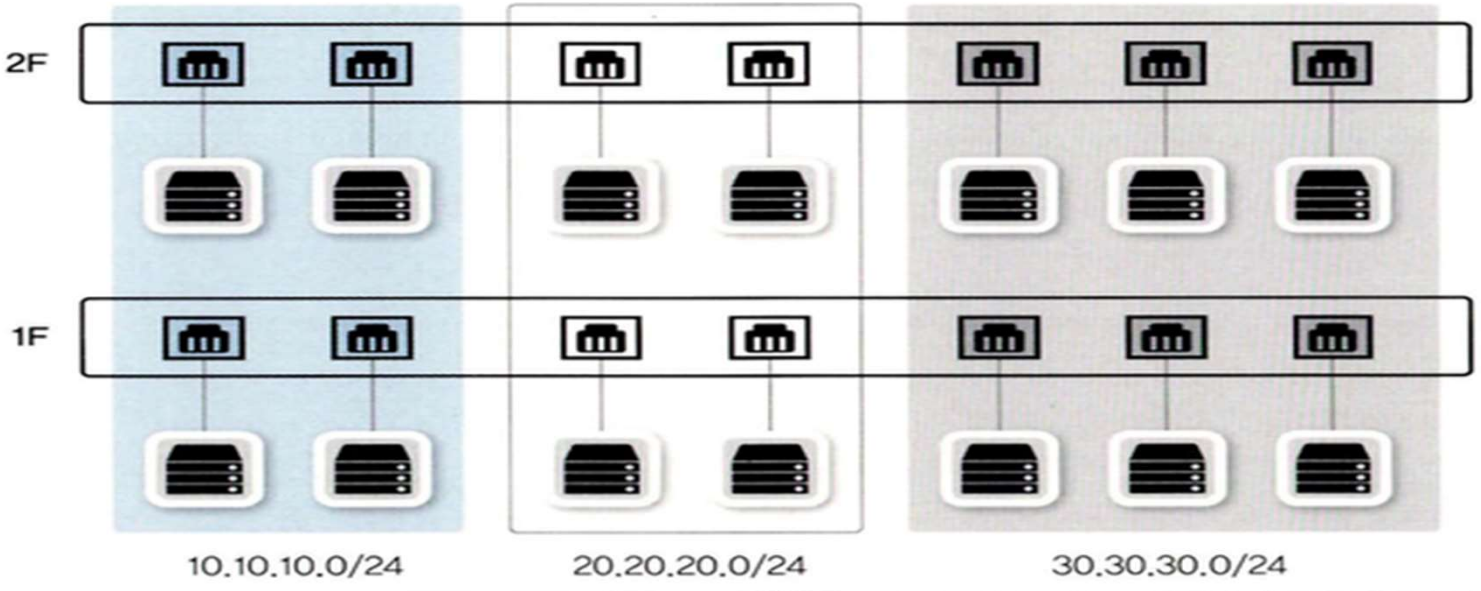


Switch

❖ VLAN

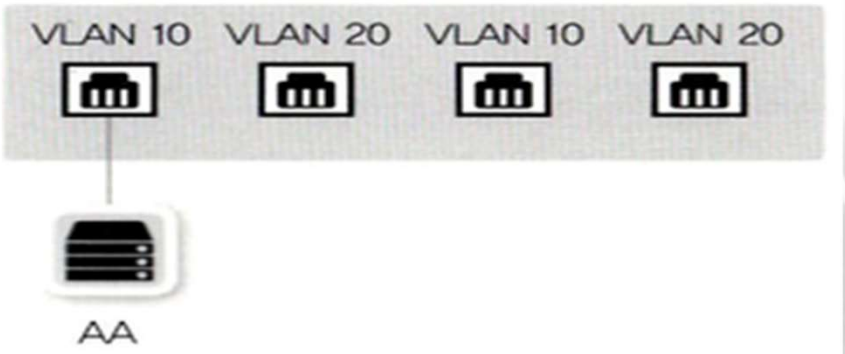
✓ 개요

- VLAN을 사용하면 물리적 구성과 상관없이 네트워크를 분리할 수 있고 물리적으로 다른 층에 있는 단말이 하나의 VLAN을 사용해 동일한 네트워크로 묶을 수 있음
- 같은 층에서 부서별로 네트워크를 분리하거나 일반 PC, IP 전화기를 분리할 수 있음
- 무선 단말과 같이 서비스나 단말의 성격에 따라 네트워크를 분리할 수 있고 분리된 단말 간에는 3계층 장비를 통해 통신



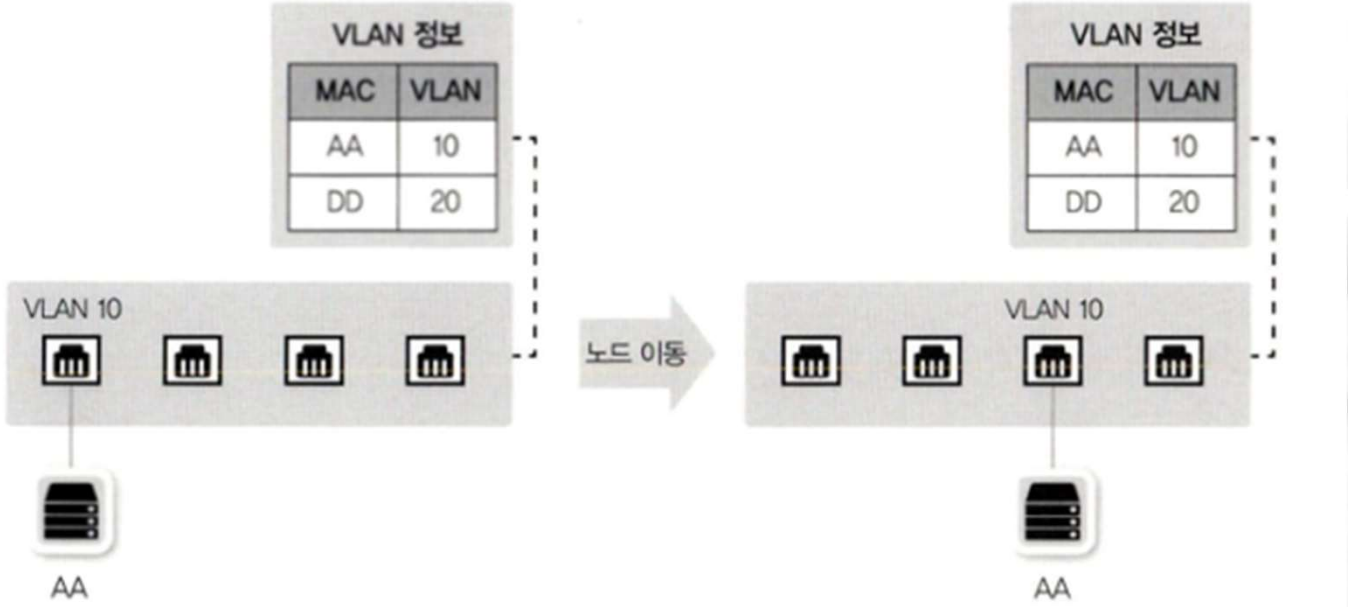
Switch

- VLAN
  - ✓ 종류 와 특징
    - VLAN 할당 방식에는 포트 기반의 VLAN과 MAC 주소 기반의 VLAN이 있음
    - VLAN 개념이 처음 도입되었을 때는 스위치가 고가였고 여러 허브를 묶는 역할을 스위치가 담당했으므로 스위치를 분할해 여러 네트워크에 사용하는 것이 VLAN 기능을 적용하는 목적
    - 포트 기반 VALN
      - ◆ 스위치를 논리적으로 분할해 사용하는 것이 목적인 VLAN을 포트 기반 VLAN(Port Based VLAN)이라고 부르고 우리가 일반적으로 언급하는 대부분의 VLAN은 포트 기반 VLAN
      - ◆ 어떤 단말이 접속하든지 스위치의 특정 포트에 VLAN을 할당하면 할당된 VLAN에 속하게 됨



Switch

- ❖ VLAN
  - ✓ 종류 와 특징
    - ● MAC 기반 VLAN
      - ◆ 사용자들의 자리 이동이 많아지면서 MAC 기반 VLAN(Mac Based VLAN)이 개발되었는데 스위치의 고정 포트에 VLAN을 할당하는 것이 아니라 스위치에 연결되는 단말의 MAC 주소를 기반으로 VLAN을 할당하는 기술
      - ◆ 단말이 연결되면 단말의 MAC 주소를 인식한 스위치가 해당 포트를 지정된 VLAN으로 변경하는데 단말에 따라 VLAN 정보가 바뀔 수 있어 Dynamic VLAN 이라고도 부름

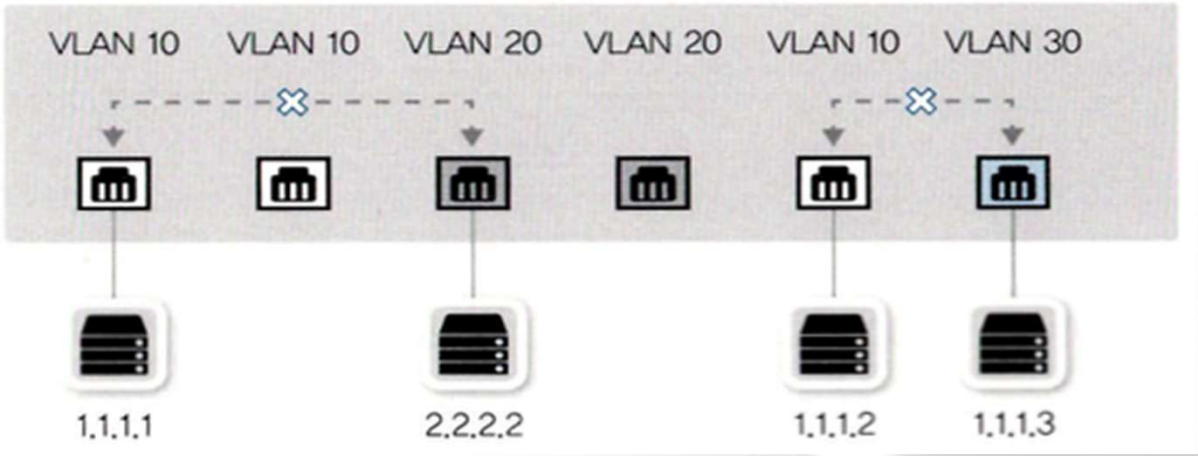


Switch

- VLAN
  - ✓ 종류 와 특징
    - 데이터 센터에서의 스위치 VLAN은 포트 기반 VLAN으로 구성하는 것이 일반적
    - 캠퍼스 네트워크에서도 포트 기반 VLAN 구성이 일반적이지만 사용자의 이동성을 요구하는 최근 스마트 오피스의 요구로 MAC 기반 VLAN 구성이 점점 더 많이 사용되고 있음

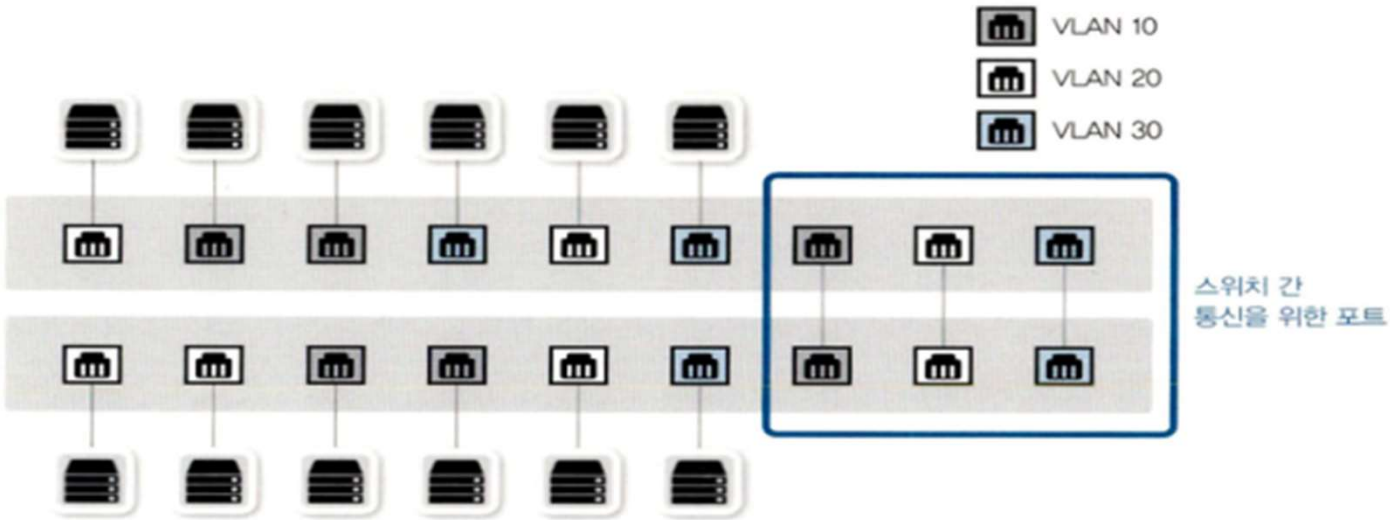
Switch

- VLAN
  - ✓ VLAN Mode 동작 방식
    - 포트 기반 VLAN에서는 스위치의 각 포트에 각각 사용할 VLAN을 설정하는데 한 대의 스위치에 연결되더라도 서로 다른 VLAN이 설정된 포트 간에는 통신할 수 없음
    - 별도의 분리된 스위치에 연결된 것과 같으므로 VLAN 간 통신이 불가능
    - 서로 다른 VLAN간 통신을 위해서는 3계층 장비를 사용해야 함
    - VLAN으로 구분된 네트워크에서는 브로드캐스트인 ARP 리퀘스트가 다른 VLAN으로 전달될 수 없으므로 3계층 장비를 이용해 통신을 해야 함



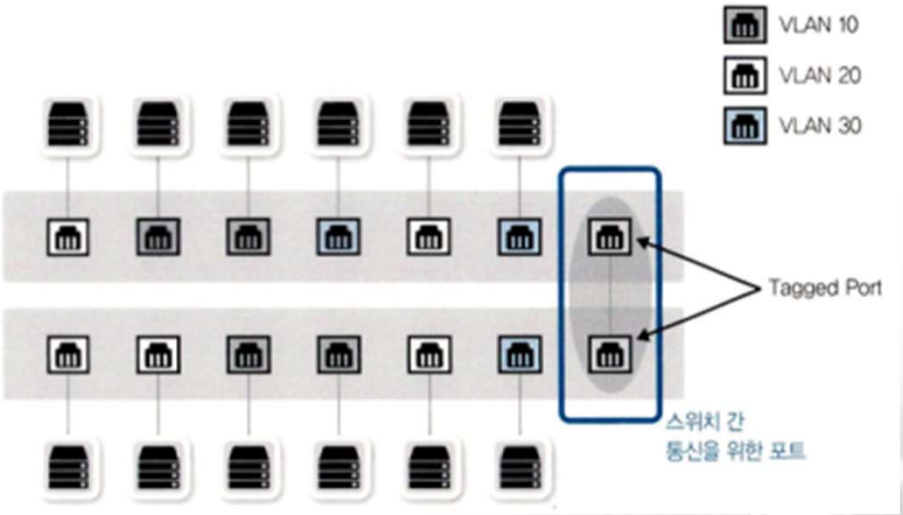
Switch

- VLAN
  - ✓ VLAN Mode 동작 방식
    - 여러 개의 VLAN이 존재하는 상황에서 스위치를 서로 연결해야 하는 경우에는 각 VLAN끼리 통신하려면 VLAN 개수만큼 포트를 연결해야 함
    - VLAN으로 분할된 스위치는 물리적인 별도의 스위치처럼 취급
    - 스위치에 3개의 VLAN이 구성되어 있는 경우 각 VLAN이 스위치 간에 통신하려면 3개의 포트가 필요
    - VLAN을 더 많이 사용하는 대형 네트워크에서 이렇게 VLAN별로 포트를 연결하면 장비 간의 연결만으로도 많은 포트가 낭비



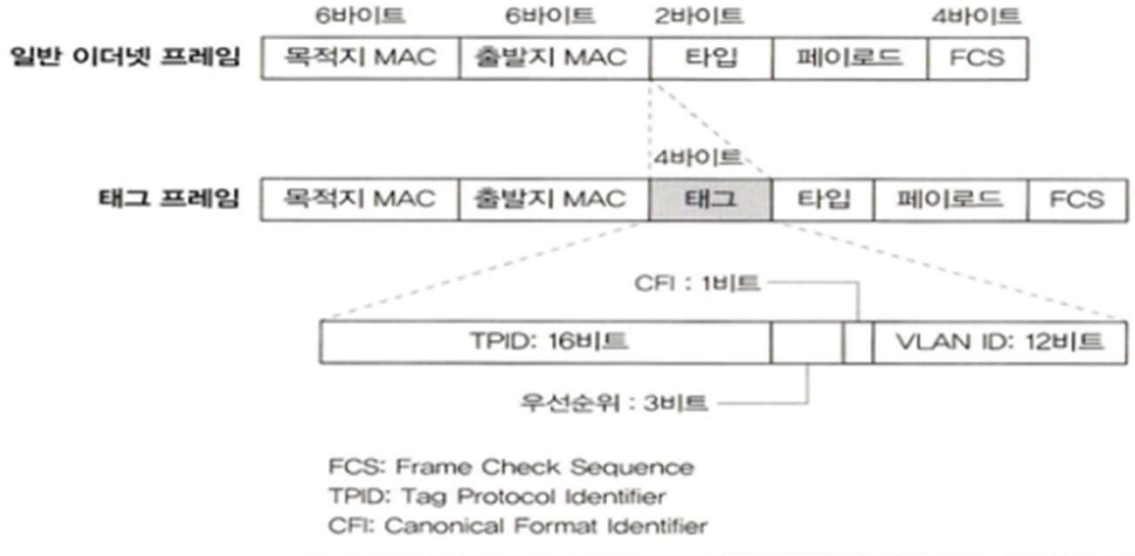
Switch

- VLAN
  - ✓ VLAN Mode 동작 방식
    - 태그 기능은 하나의 포트에 여러 개의 VLAN을 함께 전송할 수 있게 해줌
    - 이 포트를 태그(Tagged) 포트 또는 트렁크(Trunk) 포트라고 함
    - 여러 개의 VLAN을 동시에 전송해야 하는 태그 포트는 통신할 때 이더넷 프레임 중간에 VLAN ID 필드를 끼워 넣어 이 정보를 이용하는데 태그 포트로 패킷을 보낼 때는 VLAN ID를 붙이고 수신 측에서는 이 VLAN ID를 제거하면서 VLAN ID의 VLAN으로 패킷을 보낼 수 있게 됨



Switch

- VLAN
  - ✓ VLAN Mode 동작 방식
    - Trunk
      - ◆ 스위치 간 VLAN 정보를 보낼 수 있는 포트의 일반적인 용어는 태그 포트
      - ◆ 트렁크 포트는 시스코에서 사용하는 명칭으로 트렁크라는 용어를 다른 네트워크 장비 제조사에서는 다른 의미로 사용
      - ◆ 타 제조사에서는 트렁크 포트를 여러 개의 포트를 묶어 사용하기 위한 링크 애그리게이션의 의미로 사용
    - 일반 이더넷 프레임 과 태그 프레임 과의 차이



Switch

- VLAN
  - ✓ VLAN Mode 동작 방식
    - VLAN이 있는 상태에서 스위치의 MAC 테이블
      - 태그 포트를 사용하면 VLAN마다 통신하기 위해 필요했던 여 러 개의 포트를 하나로 묶어 사용할 수 있으므로 포트 낭비 없이 네트워크를 더 유연하게 디자인할 수 있음
      - 태그 포트 기능이 스위치에 생기면서 스위치의 패킷 전송에 사용하는 MAC 주소 테이블에도 변화가 생김
      - 다른 VLAN끼리 통신하지 못하도록 MAC 테이블에 VLAN을 지정하는 필드가 추가된 것
      - 하나의 스위치에서 VLAN을 이용해 네트워크를 분리하면 VLAN별로 MAC 주소 테이블이 존재하는 것처럼 동작
      - 일반적인 포트를 언태그(Untagged) 포트 또는 액세스(Access) 포트라고 하고 VLAN 정보를 넘겨 여러 VLAN이 한꺼번에 통신하도록 해주는 포트를 태그 포트 또는 트렁크 포트 라고 부름
      - 태그 포트는 여러 개의 VLAN. 즉 여러 네트워크를 하나의 물리적 포트로 전달하는 데 사용되고 언태그 포트는 하나의 VLAN에 속한 경우에만 사용
      - 일반적으로 태그 포트는 여러 네트워크가 동시에 설정된 스위치 간의 연결에서 사용되며 하나의 네트워크에 속한 서버의 경우에는 언태그로 설정

Switch

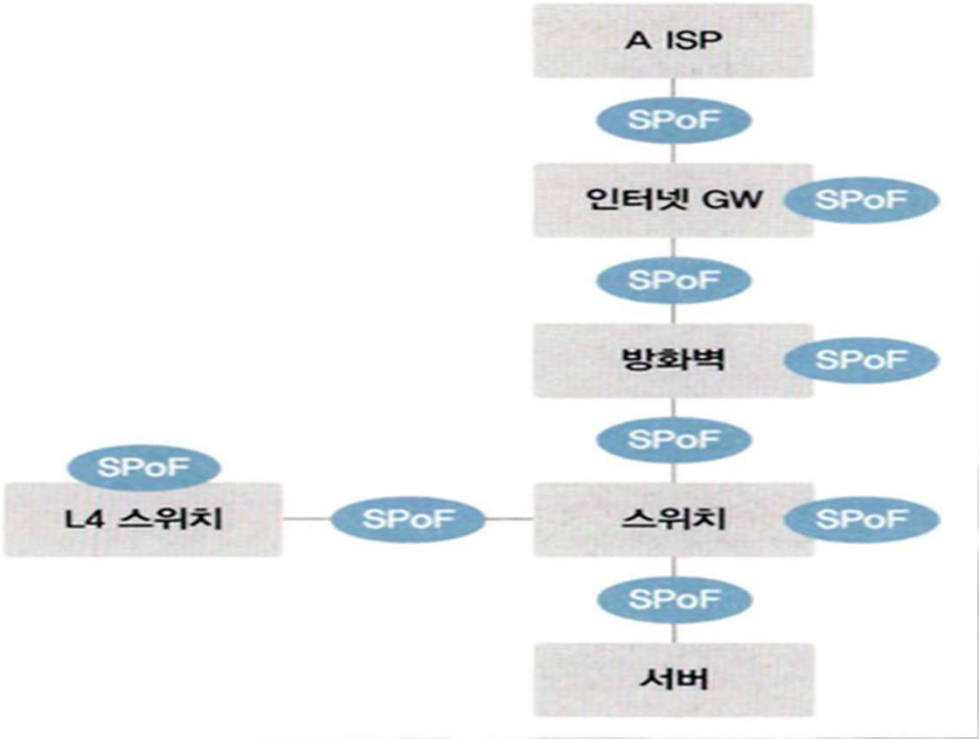
- VLAN
  - ✓ VLAN Mode 동작 방식
    - VLAN이 있는 상태에서 스위치의 MAC 테이블
      - 언태그 포트로 패킷이 들어올 경우 같은 VLAN으로만 패킷을 전송하고 태그 포트로 패킷이 들어올 경우 태그를 벗겨내면서 태그된 VLAN 쪽으로 패킷을 전송
      - 스위치 간의 연결이 아닌 서버와 연결된 포트도 VMware의 ESXi와 같은 가상화 서버가 연결될 때는 여러 VLAN과 통신해야 할 수도 있는데 이 경우 서버와 연결된 스위치의 포트더라도 언태그 포트가 아닌 태그로 설정
      - 물론 태그된(Tagged) 상태이므로 가상화 서버쪽 인터페이스에서도 태그된(Tagged) 상태로 설정해야 함
      - 가상화 서버 내부에 가상 스위치가 존재하므로 스위치 간 연결로 보면 이해하기 더 쉬움

Switch

❖ STP

✓ SPoF(Single Point of Failure – 단일 장애점)

- SPOF란 하나의 시스템이나 구성 요소에서 고장이 발생했을 때 전체 시스템의 작동이 멈추는 요소
- 네트워크에서도 하나의 장비 고장으로 전체 네트워크가 마비되는 것을 막기 위해 이중화 및 다중화된 네트워크를 디자인하고 구성

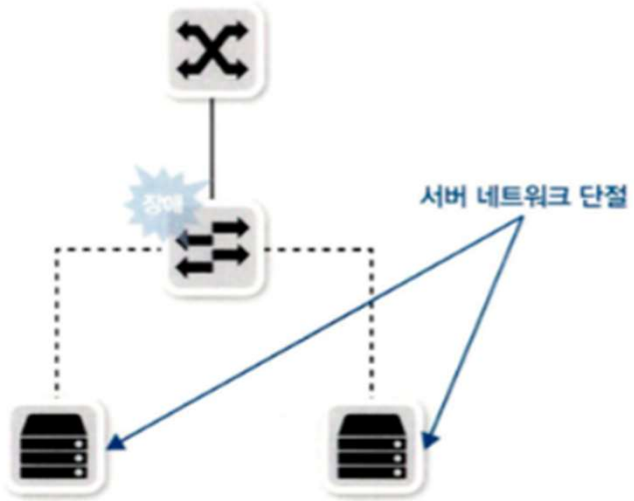


Switch

❖ STP

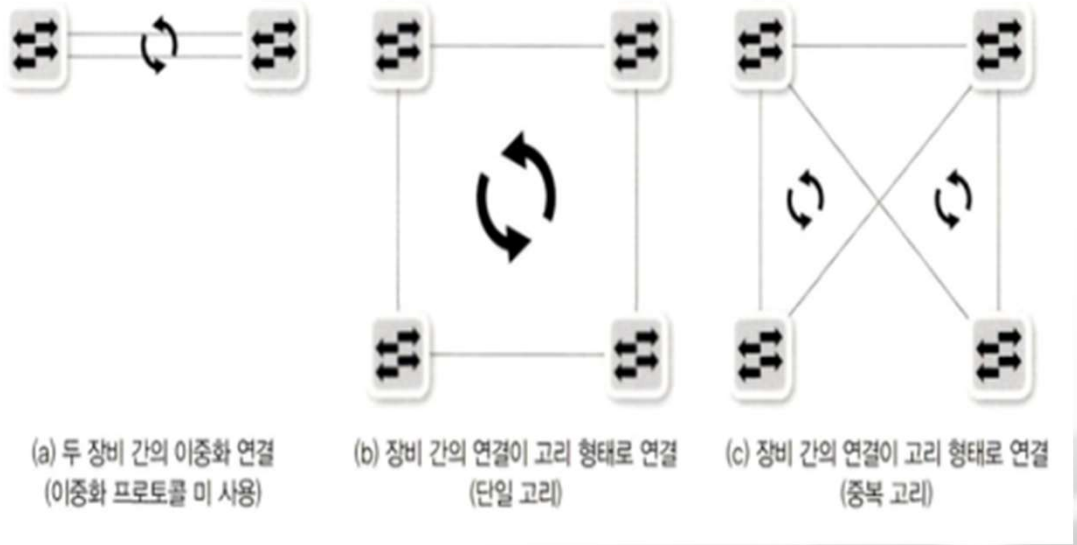
✓ SPoF(Single Point of Failure – 단일 장애점)

- 네트워크를 스위치 하나로 구성했을 때 그 스위치에 장애가 발생하면 전체 네트워크에 장애가 발생



Switch

- ❖ STP
  - ✓ Loop
    - SPoF를 피하기 위해 스위치 두 대로 네트워크를 디자인하지만 두 대 이상의 스위치로 디자인하면 패킷이 네트워크를 따라 계속 전송되므로 네트워크를 마비시킬 수 있음
    - 다양한 루프 구조

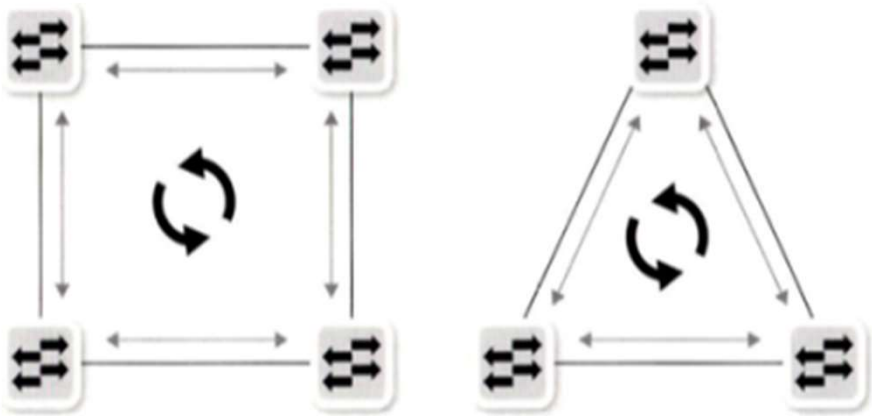


Switch

- ❖ STP
  - ✓ Loop
    - ● Broadcast Storm
      - ◆ 루프 구조로 네트워크가 연결된 상태에서 단말에서 브로드캐스트를 발생시키면 스위치는 이 패킷을 패킷이 유입된 포트를 제외한 모든 포트로 플러딩
      - ◆ 플러딩된 패킷은 다른 스위치로도 보내지고 이 패킷을 받은 스위치는 패킷이 유입된 포트를 제외한 모든 포트로 다시 플러딩
      - ◆ 루프 구조 상태에서는 이 패킷이 계속 돌아가는데 이것을 브로드캐스트 스톰이라고 함
      - ◆ 3계층 헤더에는 TTL(Timc to Live)이라는 패킷 수명을 갖고 있지만 스위치가 확인하는 2계층 헤더에는 이런 3계층의 TTL과 같은 라이프타임 메커니즘이 없어 루프가 발생하면 패킷이 죽지 않고 계속 살아남아 패킷 하나가 전체 네트워크 대역폭을 차지할 수 있음
      - ◆ 브로드캐스트 스톰은 네트워크의 전체 대역폭을 차지하고 네트워크에 연결된 모든 단말이 브로드캐스트를 처리하기 위해 시스템 리소스를 사용하면서 스위치와 네트워크에 연결된 단말 간 통신이 거의 불가능한 상태가 됨

Switch

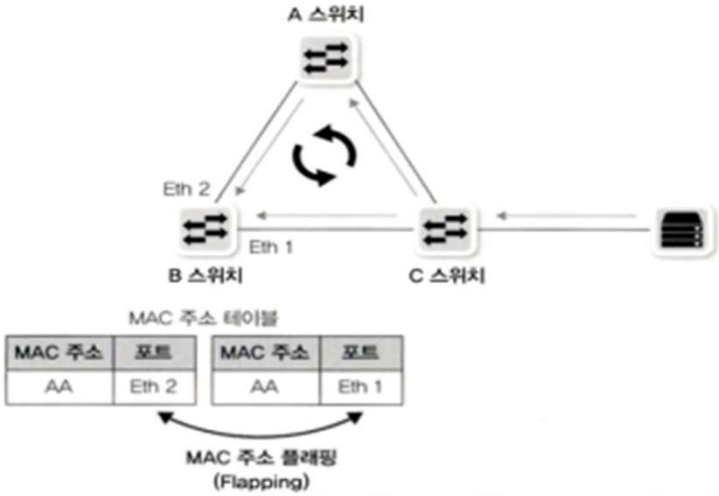
- ❖ STP
  - ✓ Loop
    - ● Broadcast Storm
      - ◆ 브로드캐스트 스톰이 발생하는 구조



      - ◆ 브로드캐스트 스톰 상황이 발생했을 때 결과
        - ▪ 네트워크에 접속된 단말의 속도가 느려짐(많은 브로드캐스트를 처리해야 하므로 CPU 사용률이 높아짐)
        - ▪ 네트워크 접속 속도가 느려짐(거의 통신 불가능 수준)
        - ▪ 네트워크에 설치된 스위치에 모든 LED들이 동시에 빠른 속도로 깜빡

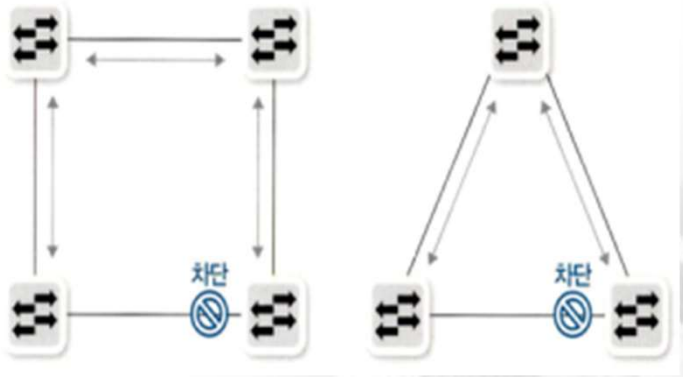
Switch

- ❖ STP
  - ✓ Loop
    - ● 스위치 MAC 러닝 중복 문제
      - ◆ 루프 구조 상태에서는 브로드캐스트 뿐 만 아니라 유니캐스트도 문제를 일으킴
      - ◆ 같은 패킷이 루프를 돌아 도착지 쪽에서 중복 수신되는 혼란을 일으키기도 하지만 중간에 있는 스위치에서도 MAC 러닝 문제가 발생
      - ◆ 스위치는 출발지 MAC 주소를 학습하는데 직접 전달되는 패킷과 스위치를 돌아 들어간 패킷 간의 포트가 달라 MAC 주소를 정상적으로 학습할 수 없음
      - ◆ 스위치 MAC 주소 테이블에서는 하나의 MAC 주소에 대해 하나의 포트만 학습할 수 있으므로 동일한 MAC 주소가 여러 포트에서 학습되면 MAC 테이블이 반복 갱신되어 정상적으로 동작하지 않는데 이 현상을 MAC 어드레스 플래핑(MAC Address Flapping) 이라고 부름



Switch

- ❖ STP
  - ✓ Loop
    - ● 스위치 MAC 러닝 중복 문제
      - ◆ 이런 현상이 발생하면 스위치에서 학습된 주소의 포트가 계속 변경되므로 스위치가 정상적으로 동작하지 못하고 패킷을 플러딩
      - ◆ 이런 현상을 예방하기 위해 스위치 설정에 따라 경고(Warning) 메시지를 관리자에게 알려주거나 수시로 일어나는 플래핑 현상을 학습하지 않도록 자동으로 조치함
      - ◆ 네트워크에 루프가 발생할 경우 앞의 문제들 때문에 네트워크가 정상적으로 동작하지 않으므로 루프가 생기지 않도록 미리 네트워크에 조치를 해야 함
      - ◆ 루프 구성 포트 중 하나의 포트만 사용하도록 셧다운(Shutdown)되어 있어도 루프를 예방할 수 있음



Switch

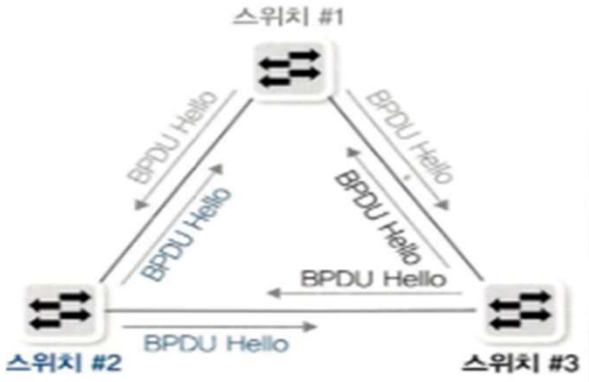
- ❖ STP
  - ✓ Loop
    - ● 스위치 MAC 러닝 중복 문제
      - ◆ 네트워크의 SPoF를 예방하기 위해 스위치를 두 개 이상 디자인했는데 다시 수동으로 루프를 찾아 강제로 사용하지 못하게 하는 방법은 바람직하지 않음
      - ◆ 먼저 네트워크에서 복잡한 케이블 연결을 이용해 루프를 찾아내는 것이 힘듬
      - ◆ 찾아내 강제로 포트를 사용하지 못하게 하더라도 네트워크에 장애가 발생하면 해당 포트를 수동으로 다시 사용하도록 해야 함
      - ◆ 사용자가 이렇게 적극적으로 개입하는 방법으로는 네트워크 장애에 적절히 대응할 수 없음
      - ◆ 이런 이유로 루프를 자동 감지해 포트를 차단하고 장애 때문에 우회로가 없을 때 차단된 포트를 스위치 스스로 다시 풀어주는 스패닝 트리 프로토콜이 개발됨

Switch

❖ STP

✓ STP 개요

- 스패닝 트리 프로토콜(Spanning Tree Protocol)은 루프를 확인하고 적절히 포트를 사용하지 못하게 만들어 루프를 예방하는 메커니즘
- 루프가 생기지 않도록 유지하는 것이 스패닝 트리 프로토콜의 목적
- 스패닝 트리 프로토콜을 이용해 루프를 예방하려면 전체 스위치가 어떻게 연결되는지 알아야 하는데 전체적인 스위치 연결 상황을 파악하려면 스위치 간에 정보를 전달하는 방법이 필요한데 이를 위해 스위치는 BPDU(Bridge Protocol Data Unit)라는 프로토콜을 통해 스위치 간에 정보를 전달하고 이렇게 수집된 정보를 이용해 전체 네트워크 트리를 만들어 루프 구간을 확인
- BPDU에는 스위치가 갖고 있는 ID와 같은 고유값이 들어가고 이런 정보들이 스위치 간에 서로 교환되면서 루프 파악이 가능한데 이렇게 확인된 루프 지점을 데이터 트래픽이 통과하지 못하도록 차단해 루프를 예방



Switch

- STP
  - ✓ 스위치 포트의 상태 및 변경 과정
    - 스패닝 트리 프로토콜이 동작 중인 스위치에서는 루프를 막기 위해 스위치 포트에 신규 스위치가 연결되면 바로 트래픽이 흐르지 않도록 차단하고 해당 포트로 트래픽이 흘러도 되는지 확인하기 위해 BPDU를 기다려 학습하고 구조를 파악한 후 트래픽을 흘리거나 루프 구조인 경우 차단 상태를 유지
    - 차단 상태에서 트래픽이 흐를 때까지 스위치 포트의 상태는 다음 4가지로 구분
      - Blocking
        - 패킷 데이터를 차단한 상태로 상대방이 보내는 BPDU를 기다림
        - 총 20초인 Max Age 기간 동안 상대방 스위치에서 BPDU를 받지 못했거나 후순위 BPDU를 받았을 때 포트는 리스닝 상태로 변경
        - BPDU 기본 교환 주기는 2초이고 10 번의 BPDU를 기다림
      - Listening: 리스닝 상태는 해당 포트가 전송 상태로 변경되는 것을 결정하고 준비하는 단계로 이 상태부터는 자신의 BPDU 정보를 상대방에게 전송하기 시작하는데 15초 동안 대기
      - Learning: 러닝 상태는 이미 해당 포트를 포워딩하기로 결정하고 실제로 패킷 포워딩이 일어날 때 스위치가 곧바로 동작하도록 MAC 주소를 러닝하는 단계로 15초 동안 대기
      - Forwarding: 패킷을 포워딩하는 단계로 정상적인 통신이 가능

Switch

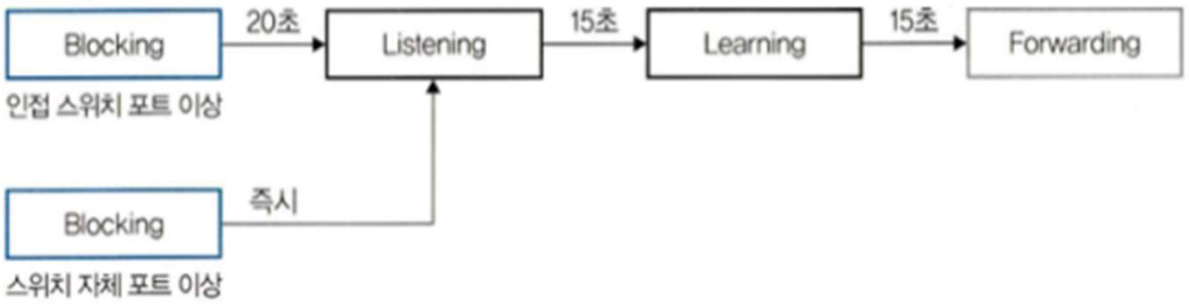
- STP
  - ✓ 스위치 포트의 상태 및 변경 과정
    - 스위치에 신규로 장비를 붙이면 통신하는 데 50여 초가 소요
    - 스위치는 루프를 예방하기위해 매우 방어적으로 동작하는데 새로 연결된 단말이 스위치일 가능성이 있어 BPDU를 일정 시간 이상 기다려 스위치 여부를 파악하는데 이로 인해 스위치를 연결하는 경우뿐만 아니라 일반 단말을 연결하더라도 동일한 시간이 필요
    - 이중화된 링크 절체(전환)도 STP의 동작 순서대로 진행

Switch

❖ STP

✓ 스위치 포트의 상태 및 변경 과정

- 특정 링크가 다운되어 블로킹 포트가 포워딩되기 위해 초기와 마찬가지로 20초 동안 Max Age를 거쳐 총 50초 후 포워딩 상태로 변경되지만 다운된 링크가 자신의 인터페이스인 경우 토폴로지가 변했음을 직접 감지할 수 있어 Max Age를 거치지 않고 리스닝부터 STP 상태 변화가 즉시 이루어지므로 30초 만에 전환
- STP가 활성화된 경우 스위치 포트는 곧바로 포워딩 상태가 되지 않기 때문에 다양한 장애가 발생하거나 스위치 이상으로 생각되는 경우가 많음
- 부팅 시간이 매우 빠른 OS가 DHCP 네트워크에 접속할 때 부팅 단계에서 IP를 요청하지만 스위치 포트가 포워딩 상태가 되지않아 IP를 정상적으로 할당받지 못하는 경우가 많음



Switch

- ❖ STP
  - ✓ STP 동작 방식
    - STP는 루프를 없애기 위해 나무가 뿌리에서 가지로 뻗어나가는 것처럼 토폴로지를 구성
    - 네트워크상에서 뿌리가 되는 가장 높은 스위치를 선출하고 그 스위치를 통해 모든 BPDU가 교환되도록 하는데 그 스위치가 루트 스위치
    - 모든 스위치는 처음에 자신을 루트 스위치로 인식해 동작
    - BPDU를 통해 2초마다 자신이 루트 스위치임을 광고하는데 새로운 스위치가 들어오면 서로 교환된 BPDU에 들어 있는 브릿지 ID 값을 비교하고 브릿지 ID 값이 더 적은 스위치를 루트 스위치로 선정하고 루트 스위치로 선정된 스위치가 BPDU를 다른 스위치 쪽으로 보냄

Switch

- ❖ STP
  - ✓ STP 동작 방식
    - ● 스패닝 트리 프로토콜은 루프를 예방하기 위해 다음과 같이 동작
      - ◆ 하나의 루트 스위치를 선정
        - ▪ 전체 네트워크에 하나의 루트 스위치를 선정
        - ▪ 자신을 전체 네트워크의 대표 스위치로 적은 BPDU를 옆 스위치로 전달
      - ◆ 루트가 아닌 스위치 중 하나의 루트 포트를 선정
        - ▪ 루트 브릿지로 가는 경로가 가장 짧은 포트를 루트 포트라고 함
        - ▪ 루트 브릿지에서 보낸 BPDU를 받는 포트
      - ◆ 하나의 세그먼트에 하나의 지정(Designated) 포트를 선정
        - ▪ 스위치와 스위치가 연결되는 포트는 하나의 지정 포트(Designated Pori)를 선정합니다
        - ▪ 스위치 간의 연결에서 이미 루트 포트로 선정된 경우 반대쪽이 지정 포트로 선정되어 양쪽 모두 포워딩 상태가 됨
        - ▪ 스위치 간의 연결에서 아무도 루트 포트가 아닐 경우 한쪽은 지정 포트로 선정되고 다른 한쪽은 대체 포트(alternate, non-designated)가 되어 차단 상태가 됨
        - ▪ BPDU가 전달되는 포트

Switch

- ❖ STP
  - ✓ STP 동작 방식
    - ● 스패닝 트리 프로토콜 사용 시 대안(Port Fast)
      - ◆ 포트에 새로운 케이블이 연결되면 곧바로 포워딩 상태로 변하는 것이 아니라 상대방이 스위치일 수도 있다고 가정하고 BPDU가 들어오는지 모니터링하지만 이런 메커니즘은 단말이 네트워크에 연결될 때까지 시간이 지연되는 문제 때문에 스위치가 아닌 일반 PC나 서버가 연결되는 포트라면 이런 메커니즘을 아예 없애거나 좀 더 빠른 시간 안에 포워딩 상태로 변경되어야 하는데 이런 경우 해당 포트를 포트 패스트(Port Fast)로 설정하면 BPDU 대기, 습득 과정 없이 곧바로 포워딩 상태로 포트를 사용할 수 있ㅇ,ㅁ
      - ◆ 포트 패스트를 설정한 포트에 스위치가 접속되면 루프가 생길 수 있어 별도로 해당 포트에 BPDU가 들어오자마자 포트를 차단하는 BPDU 가드(Guard) 같은 기술이 함께 사용되어야 함

Switch

- ❖ STP
  - ✓ 향상된 STP
    - ● RSTP
      - ◆ 스패닝 트리 프로토콜은 이중화된 스위치 경로 중 정상적인 경로에 문제가 발생할 경우 백업 경로를 활성화하는 데 30~50초가 걸림
      - ◆ 백업 경로를 활성화하는 데 시간이 너무 오래 걸리는 문제를 해결하기 위해 RSTP(Rapid Spanning Tree Protocol)가 개발됨
      - ◆ RSTP는 2~3초로 전환 시간이 짧아 일반적인 TCP 기반 애플리케이션이 세션을 유지할 수 있게 됨
      - ◆ 기본적인 구성과 동작 방식은 STP와 같지만 BPDU 메시지 형식이 다양해져 여러 가지 상태 메시지를 교환할 수 있음
      - ◆ STP는 일반 토폴로지 변경과 관련된 두 가지 메시지(TCN, TCA BPDU)만 있지만 RSTP는 8개 비트를 모두 활용해 다양한 정보를 주위 스위치와 주고받을 수 있음
      - ◆ 기존 STP에서는 토폴로지가 변경되면 말단 스위치에서 루트 브릿지까지 변경 보고를 보내고 루트 브릿지가 그에 대한 연산을 다시 완료하고 이후 변경된 토폴로지 정보를 말단 스위치까지 보내는 과정을 거치고 추가로 이런 정보가 네트워크에 있는 모든 스위치까지 전파되는 예비시간까지 고려해야 하므로 정보를 확정하는 데 시간이 오래 걸림
      - ◆ RSTP에서는 토폴로지 변경이 일어난 스위치 자신이 모든 네트워크에 토폴로지 변경을 직접 전파할 수 있음

Switch

- ❖ STP
  - ✓ 향상된 STP
    - ● MST
      - ◆ 일반 스패닝 트리 프로토콜은 VLAN 개수와 상관없이 스패닝 트리 한 개만 동작하는데 이 경우 VLAN이 많더라도 스패닝 트리는 한 개만 동작하면 되므로 스위치의 관리 부하가 적지만 루프가 생기는 토폴로지에서 한 개의 포트와 회선만 활성화되므로 자원을 효율적으로 활용할 수 없음
      - ◆ VLAN마다 최적의 경로가 다를 수 있는데 포트 하나만 사용할 수 있다보니 멀리 돌아 통신해야 할 경우도 생김
      - ◆ 이 문제를 해결하기 위해 PVST(Per Vlan Spanning Tree)가 개발되었고 VLAN마다 다른 스패닝 트리 프로세스가 동작하므로 VLAN마다 별도의 경로와 트리를 만들 수 있게 되는데 최적의 경로를 디자인하고 VLAN마다 별도의 블록 포트를 지정해 네트워크 로드를 셰어링하도록 구성할 수 있게 됨
      - ◆ 스패닝 트리 프로토콜 자체가 스위치에 많은 부담을 주는 프로토콜인데 PVST는 모든 VLAN마다 별도의 스패닝 트리를 유지해야 하므로 더 많은 부담이 됨
      - ◆ PVST의 단점을 보완하기 위해 MST(Multipk- Spanning Tree)가 개발됨

Switch

- ❖ STP
  - ✓ 향상된 STP
    - ● MST
      - ◆ MST의 기본적인 아이디어는 매우 단순한데 여러 개의 VLAN을 그룹으로 묶고 그 그룹마다 별도의 스패닝 트리가 동작하는 방식인데 이 경우 PVST보다 훨씬 적은 스패닝 트리 프로토콜 프로세스가 돌게 되고 PVST의 장점인 로드 셰어링 기능도 함께 사용할 수 있음
      - ◆ 일반적으로 대체 경로의 개수나 용도에 따라 MST의 스패닝 트리 프로토콜 프로세스 개수를 정의하는데 MST에서는 리전 개념이 도입되어 여러 개의 VLAN을 하나의 리전으로 묶고 리전 하나가 스패닝 트리 하나가 됨
      - ◆ 11 ~ 50번 VLAN과 101 ~ 150번 VLAN이 11~ 50번을 하나의 리전으로 101~ 150번을 하나의 리전으로 묶으면 두 개의 스패닝 트리로 100개의 VLAN을 관리할 수 있음

Switch

- STP
  - ✓ 스패닝 트리 프로토콜의 대안
    - 스패닝 트리 프로토콜은 루프 예방에 필수적인 메커니즘이지만 스위치에 부담이 많은 프로토콜이고 포트가 차단 되거나 늦게 포워딩되어 불편한 점이 많은데 이런 문제들을 해결하기 위해 PortFast, UplinkFast, BackboneFast와 같은 다수 기능들이 있지만 잘못 사용하면 장애 발생의 원인이 되기도 함(포트 패스트를 활성화한 포트에 일반 스위치가 접속되면 루프가 발생)
    - 여러 가지 불편한 스패닝 트리 프로토콜을 근본적으로 대체하기 위해 네트워크를 잘게 쪼개 디자인하거나 대체재를 사용하는 경우가 많음
    - 이런 대체재 프로토콜은 대부분 모두 호환되는 것이 아니라 스위치 제작업체에서만 동작하거나 이름은 같아도 동작 방식이 다르거나 호환성이 없는 경우가 있음
    - 종류
      - SLPP(Simple Loop Prevention Protocol): Nortel사가 개발한 루프 예방 프로토콜
      - Extreme STP: Extreme사가 개발한 루프 예방 프로토콜
      - Loop Guard: STP와 다른 메커니즘으로 루프를 확인
      - BPDU Guard: BPDU가 인입될 경우 해당 포트를 차단

Switch

- 스위치의 구조와 스위치에 IP 주소가 할당된 이유
  - ✓ 스위치는 스위치 관리용 컨트롤 플레인(Control Plane)과 패킷을 포워딩하는 데이터 플레인(Data Plane)으로 크게 나눔
  - ✓ STP나 스위치 원격 관리용 텔넷, SSH, 웹과 같은 서비스는 컨트롤 플레인에서 수행
  - ✓ 스위치는 2계층에서 동작하는 장비여서 MAC 주소만 이해할 수 있기 때문에 스위치가 동작하는 데 IP는 필요없지만 일정 규모 이상의 네트워크에서 운영되는 스위치는 관리 목적으로 대부분 IP 주소가 할당됨

Router/L3 Switch

❖ 개요

- ✓ 라우터(Router)는 3계층에서 동작하는 여러 네트워크 장비의 대표격으로 그 이름처럼 경로를 지정해주는 장비
- ✓ 라우터에 들어오는 패킷의 목적지 IP 주소를 확인하고 자신이 가진 경로(Route) 정보를 이용해 패킷을 최적의 경로로 포워딩
- ✓ 인터넷을 통해 다양한 서비스를 제공받는 현대 네트워크 환경에서는 인터넷을 연결하기 위한 원격지 통신이 매우 중요
- ✓ 라우터는 원격지 네트워크와 연결할 때 필수 네트워크 장비이며 네트워크를 구성하는 핵심 장비

Router/L3 Switch

❖ 개요

✓ L3 Switch

- 스위치는 대표적인 2계층 장비이지만 라우터처럼 3계층에서 동작하는 스위치
- 기존에 라우터는 소프트웨어로 구현하고 스위치는 하드웨어로 구현하는 형태로 구분하거나 다양한 기능의 라우터와 패킷을 빨리 보내는 데 최적화된 스위치로 구분했지만 최근 기술 발달로 라우터와 L3 스위치를 구분하기는 어려움

Router/L3 Switch

❖ 라우터의 동작 방식과 역할

✓ 개요

- 라우터는 다양한 경로 정보를 수집해 최적의 경로를 라우팅 테이블에 저장한 후 패킷이 라우터로 들어오면 도착지 IP 주소와 라우팅 테이블을 비교해 최선의 경로로 패킷을 내보냄
- 스위치와 반대로 라우터는 들어온 패킷의 목적지 주소가 라우팅 테이블에 없으면 패킷을 버림
- 라우터는 패킷 포워딩 과정에서 기존 2계층 헤더 정보를 제거한 후 새로운 2계층 헤더를 만들어 냄
- 라우터의 동작 방식을 경로 지정, 브로드캐스트 컨트롤, 프로토콜 변환이라 함

Router/L3 Switch

❖ 라우터의 동작 방식과 역할

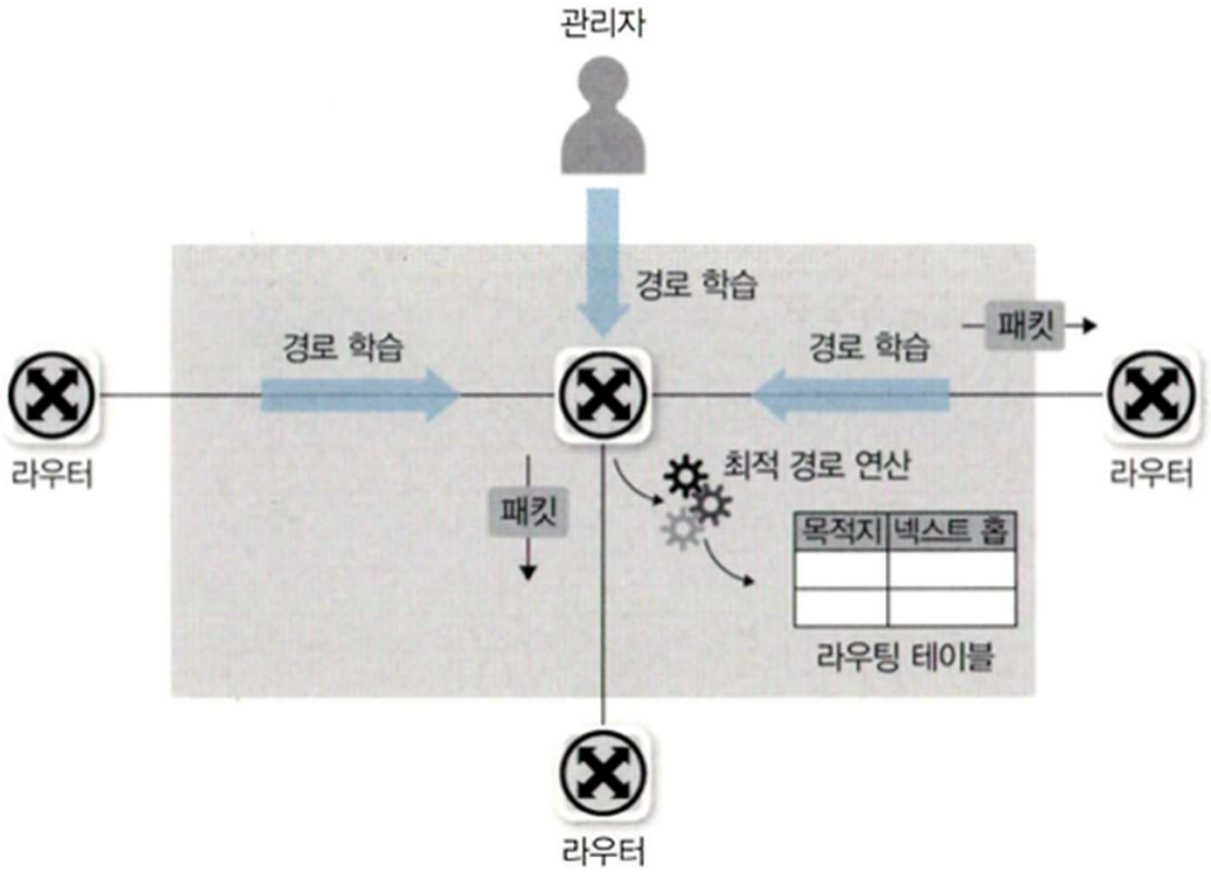
✓ 경로 지정

- 라우터의 가장 중요한 역할은 경로 지정
- 경로 정보를 모아 라우팅 테이블을 만들고 패킷이 라우터로 들어오면 패킷의 도착지 IP 주소를 확인해 경로를 지정하고 패킷을 포워딩
- IP 주소는 네트워크 주소와 호스트 주소로 나뉜 계층 구조를 기반으로 설계되어 로컬 네트워크와 원격지 네트워크를 구분할 수 있고 네트워크 주소를 기반으로 경로를 찾아 갈 수 있음
- 라우터는 이 IP 주소를 확인해 원격지에 있는 적절한 경로로 패킷을 포워딩
- 라우터는 경로를 지정해 패킷을 포워딩하는 역할을 두 가지로 구분해 수행하는데 경로 정보를 얻는 역할과 얻은 경로 정보를 확인하고 패킷을 포워딩하는 부분으로 구분
- 라우터는 자신이 얻은 경로 정보에 포함되는 패킷만 포워딩하므로 정확한 목적지 경로를 얻는 것이 매우 중요
- 다양한 방법으로 경로 정보를 얻을 수 있는데 IP 주소를 입력하면서 자연스럽게 인접 네트워크 정보를 얻는 방법과 관리자가 직접 경로 정보를 입력하는 방법 및 라우터끼리 서로 경로 정보를 자동으로 교환하는 방법이 있음

Router/L3 Switch

❖ 라우터의 동작 방식과 역할

✓ 경로 지정



Router/L3 Switch

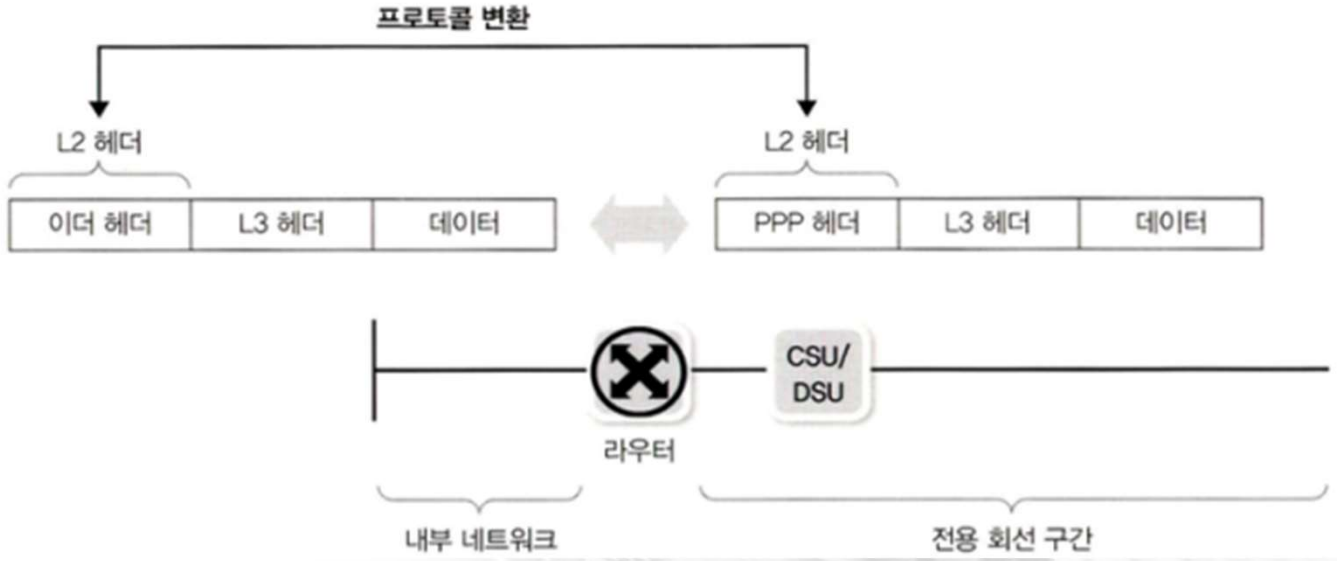
❖ 라우터의 동작 방식과 역할

✓ 브로드캐스트 컨트롤

- 스위치는 패킷의 도착지 주소를 모르면 어딘가에 존재할지 모를 장비와의 통신을 위해 플러딩해 패킷을 모든 포트에 전송하는데 LAN 어딘가에 도착지가 있을 수 있다고 가정하고 패킷을 전체 네트워크에 전송하면 쓸모없는 패킷이 전송되어 전체 네트워크의 성능에 무리가 갈 수 있다고 생각할 수 있지만 LAN은 크기가 작아 플러딩에 대한 영향이 작고 도착지 네트워크 인터페이스 카드에서 자신의 주소와 패킷의 도착지 주소가 다르면 패킷을 버리기 때문에 이런 플러딩 작업은 네트워크에 큰 무리를 주지 않음
- 라우터는 패킷을 원격지로 보내는 것을 목표로 개발되어 3계층에서 동작하고 분명한 도착지 정보가 있을 때만 통신을 허락하는데 인터넷 연결은 대부분 지정된 대역폭만 빌려 사용하므로 쓸모없는 통신이 네트워크를 차지하는 것을 최대한 막으려고 노력함
- LAN에서 스위치가 동작하는 것처럼 목적지가 없거나 명확하지 않은 패킷이 플러딩된다면 인터넷에 쓸모 없는 패킷이 가득 차 통신 불능 상태가 될 수 있음
- 라우터는 바로 연결되어 있는 네트워크 정보를 제외하고 경로 습득 설정을 하지 않으면 패킷을 포워딩할 수 없음
- 라우터의 기본 동작은 멀티캐스트 정보를 습득하지 않고 브로드캐스트 패킷을 전달하지 않는데 라우터의 이 기능을 이용해 브로드캐스트가 다른 네트워크로 전파되는 것을 막을 수 있는데 이 기능을 브로드캐스트 컨트롤/멀티캐스트 컨트롤이라고 함
- 네트워크에 브로드캐스트가 많이 발생하는 경우 라우터로 네트워크를 분리하면 브로드캐스트 네트워크를 분할해 네트워크 성능을 높일 수 있음

Router/L3 Switch

- ❖ 라우터의 동작 방식과 역할
  - ✓ 프로토콜 변환
    - 라우터의 또 다른 역할은 서로 다른 프로토콜로 구성된 네트워크를 연결하는 것
    - 현대 네트워크는 이더넷으로 수렴되므로 이 역할이 많이 줄었지만 과거에는 LAN에서 사용하는 프로토콜과 WAN에서 사용하는 프로토콜이 전혀 다른 완전히 구분된 공간 이었음
    - LAN은 다수 컴퓨터가 함께 통신하는 데 초점을 맞추었고 WAN은 원거리 통신이 목적 LAN 기술이 WAN 기술로 변환되어야만 인터넷과 같이 원격지 네트워크와의 통신이 가능했고 이 역할을 라우터가 담당



Router/L3 Switch

- ❖ 라우터의 동작 방식과 역할
  - ✓ 프로토콜 변환
    - 라우터는 3계층에서 동작하는 장비이므로 3계층 주소 정보를 확인하고 그 정보를 기반으로 동작
    - 라우터에 패킷이 들어오면 2계층까지의 헤더 정보를 벗겨내고 3계층 주소를 확인한 후 2계층 헤더 정보를 새로 만들어 외부로 내보냄
    - 라우터에 들어올 때의 패킷 2계층 헤더 정보와 나갈 때의 패킷 2계층 헤더 정보가 다름
    - 저속 전용회선에서 WAN 구간은 PPP와 같은 WAN 프로토콜이 사용되고 LAN 구간은 이더넷 프로토콜이 사용됨
    - LAN 구간에서 패킷이 라우터를 지나면서 2계층 까지의 헤더가 벗겨지고 WAN 구간으로 패킷이 나올 때 PPP 헤더로 변경되어 프로토콜이 변환됨

Router/L3 Switch

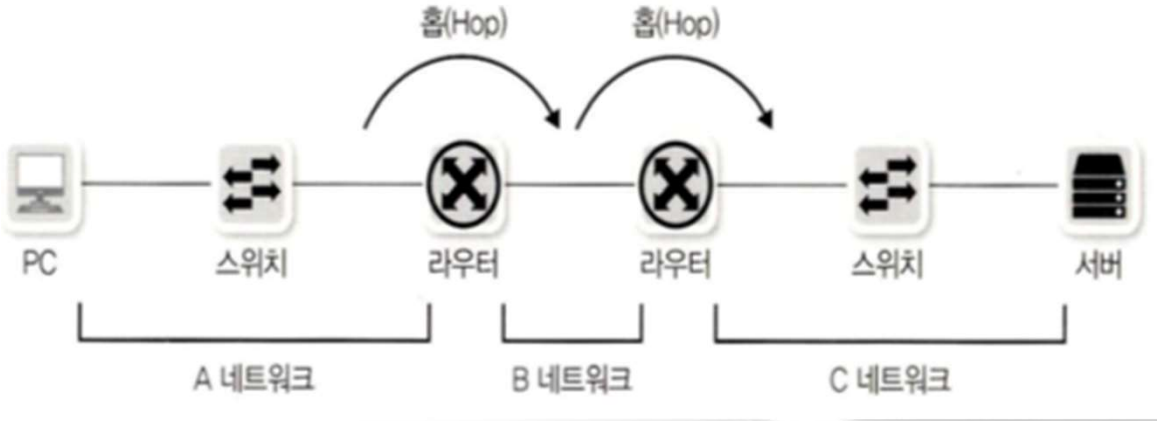
❖ 경로 지정 – 라우팅/스위칭

✓ 개요

- 라우터가 패킷을 처리할 때 수행하는 작업
  - ◆ 경로 정보를 얻어 경로 정보를 정리하는 역할
  - ◆ 정리된 경로 정보를 기반으로 패킷을 포워딩하는 역할
- 라우터는 자신이 분명히 알고 있는 주소가 아닌 목적지를 가진 패킷이 들어오면 해당 패킷을 버리므로 패킷이 들어오기 전에 경로 정보를 충분히 수집하고 있어야 라우터가 정상적으로 동작
- 인터넷에 존재하는 주소와 경로는 매우 많고 점점 늘고 있는데 클래스리스 네트워크로 전환된 후에는 같은 클래스에 있는 주소조차 서브네팅된 상태로 분산되어 존재하므로 경로 정보가 기존보다 훨씬 많아졌음
- 라우터는 이런 복잡하고 많은 경로 정보를 얻어 최적의 경로 정보인 라우팅 테이블을 적절히 유지해야 하는데 라우터는 다양하고 많은 경로 정보를 얻을 수 있지만 원하는 목적지 정보와 정확히 일치하지 않는 경우가 더 많음
- 라우터는 서브넷 단위로 라우팅 정보를 습득하고 라우팅 정보를 최적화하기 위해 서머리(Summary) 작업을 통해 여러 개의 서브넷 정보를 뭉쳐 전달
- 라우터에 들어온 패킷의 목적지 주소와 라우터가 갖고 있는 라우팅 테이블 정보가 정확히 일치하지 않더라도 수 많은 정보 중 목적지에 가장 근접한 정보를 찾아 패킷을 포워딩해야 함

Router/L3 Switch

- ❖ 경로 지정 – 라우팅/스위칭
  - ✓ 라우팅 동작 과 라우팅 테이블
    - 현대 인터넷에서는 단말부터 목적지까지의 경로를 모두 책임지는 것이 아니라 인접한 라우터까지만 경로를 지정하면 인접 라우터에서 최적의 경로를 다시 파악한 후 라우터로 패킷을 포워딩
    - 네트워크를 한 단계씩 뛰어넘는다는 의미로 이 기법을 흡-홉(Hop-by-Hop) 라우팅이라고 부르고 인접한 라우터를 넥스트 홉(Next Hop) 이라고 함
    - 라우터는 패킷이 목적지로 가는 전체 경로를 파악하지 않고 최적의 넥스트 홉을 선택해 보내줌

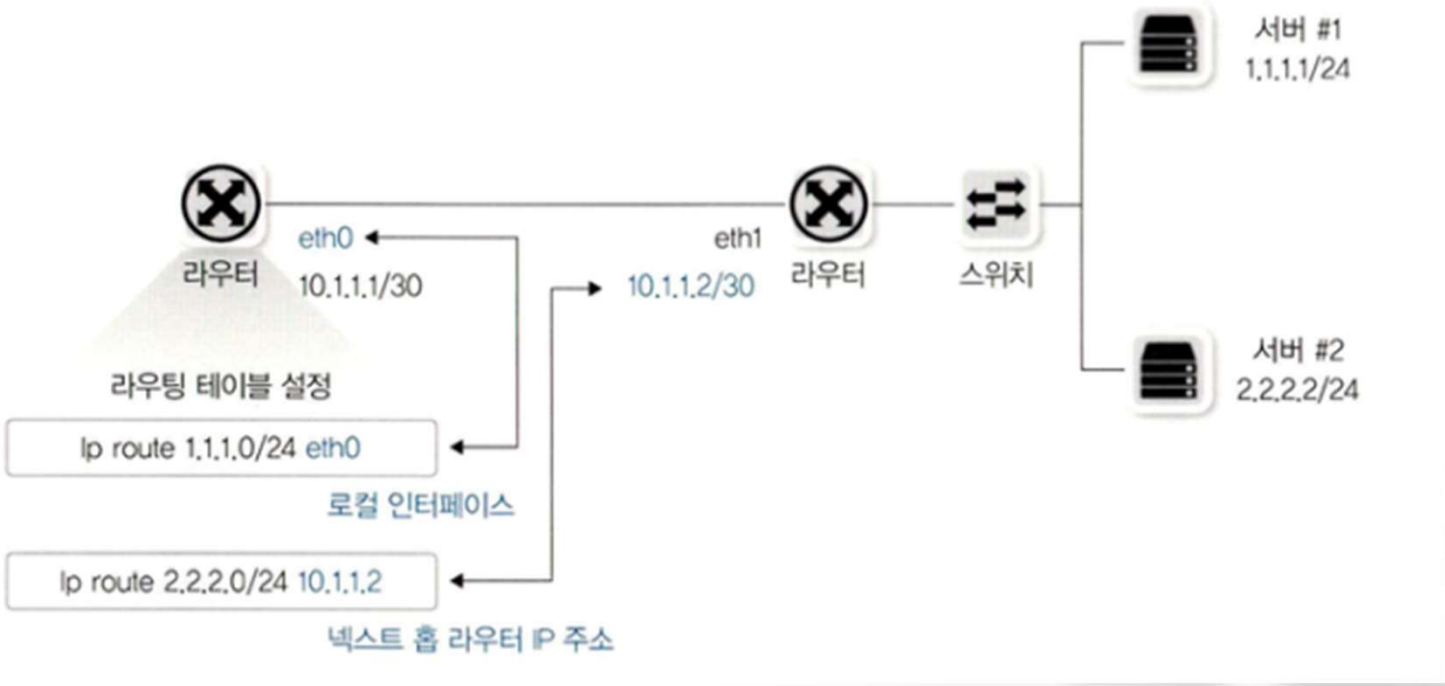


Router/L3 Switch

- 경로 지정 – 라우팅/스위칭
  - ✓ 라우팅 동작 과 라우팅 테이블
    - 넥스트 홉을 지정할 때는 일반적으로 세 가지 방법을 사용할 수 있음
      - 다음 라우터의 IP를 지정하는 방법
      - 라우터의 나가는 인터페이스를 지정하는 방법
      - 라우터의 나가는 인터페이스와 다음 라우터의 IP를 동시에 지정하는 방법
    - 라우터에서 넥스트 홉을 지정할 때는 일반적으로 상대방 라우터의 인터페이스 IP 주소를 지정하는 방법을 사용
    - 특수한 경우에만 라우터의 나가는 인터페이스를 지정하는 방법을 사용할 수 있는데 상대방 넥스트 홉 라우터의 IP를 모르더라도 MAC 주소 정보를 알아낼 수 있을 때만 사용할 수 있음
    - WAN 구간 전용선에서 PPP 나 HDLC 와 같은 프로토콜을 사용해 상대방의 MAC 주소를 알 필요가 없거나 상대방 라우터에서 프록시 ARP가 동작해 정확한 IP 주소를 모르더라도 상대방의 MAC 주소를 알 수 있을 때와 같이 한정된 조건에서만 사용할 수 있음
    - 인터페이스를 설정할 때는 라우터의 나가는 물리 인터페이스를 지정하는 것이 일반적이지만 IP주소와 인터페이스를 동시에 사용할 때는 VLAN 인터페이스와 같은 논리적인 인터페이스를 사용할 수 있음

Router/L3 Switch

- 경로 지정 – 라우팅/스위칭
  - ✓ 라우팅 동작 과 라우팅 테이블

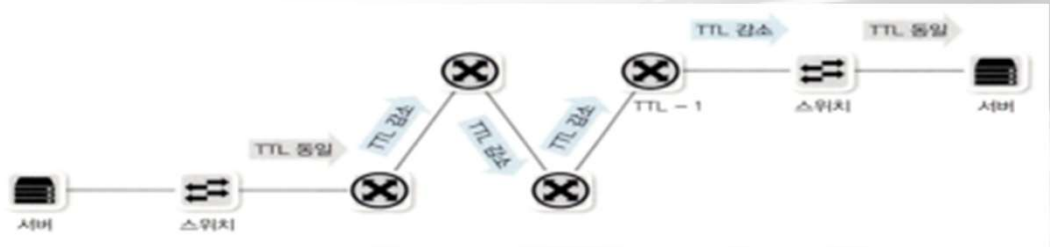


Router/L3 Switch

- 경로 지정 – 라우팅/스위칭
  - ✓ 라우팅 동작 과 라우팅 테이블
    - 라우터가 패킷을 어디로 포워딩할지 경로를 선택할 때는 출발지를 고려하지 않음
    - 출발지와 상관없이 목적지 주소와 라우팅 테이블을 비교해 어느 경로로 포워딩할지 결정
    - 라우팅 테이블을 만들 때는 목적지 정보만 수집하고 패킷이 들어오면 목적지 주소를 확인해 패킷을 넥스트 홉으로 포워딩
    - 라우팅 테이블에 저장하는 데이터에는 다음과 같은 정보를 포함
      - ◆ 목적지 주소
      - ◆ 넥스트 홉 IP 주소, 나가는 로컬 인터페이스(선택 가능)
    - 라우터에서 패킷의 출발지 주소를 이용해 라우팅하도록 PBR(Policy-Based Routing) 기능을 사용할 수 있지만 목적지 주소만 수집하는 라우팅 테이블로는 이 기능을 활성화할 수 없고 라우터 정책과 관련된 별도 설정이 필요한데 이 경우 다른 라우터로의 전파가 어렵고 라우터에 일반적이지 않은 별도 동작이 필요하며 소스 라우팅(Source Routing)이나 줄여서 폴리시 라우팅(Policy Routing)이라고 불리는 이 기능은 관리가 어려워지고 문제가 발생하면 해결이 어려우므로 특별한 목적으로만 사용

Router/L3 Switch

- ❖ 경로 지정 – 라우팅/스위칭
  - ✓ 라우팅 동작 과 라우팅 테이블
    - ● 루프가 없는 3계층: TTL(Time To Live)
      - ◆ 3계층의 ip 헤더에는 TTL이라는 필드가 있는데 이 필드는 패킷이 네트워크에 살아 있을 수 있는 시간(홉)을 제한
      - ◆ 인터넷 구간에서 쓸모없는 패킷이 돌아다녀 대역폭을 낭비하는 것을 막기 위해 라우터는 주소가 불분명한 패킷을 버리는데 하지만 실제로 운영되던 사이트가 갑자기 없어지는 경우가 발생할 수 있고 대안 경로를 찾다가 순간적으로 마주보는 두 대의 라우터의 넥스트 홉이 각각 상대방으로 구성되어 패킷이 두 라우터 사이에서 계속 오가는 경우도 생길 수 있음
      - ◆ 패킷이 영구적으로 사라지지 않는다면 장비 간에 동일한 패킷이 핑퐁(Ping Pong)을 치거나 인터넷에 사라지지 않는 유령 패킷이 넘쳐날 것
      - ◆ 모든 패킷은 TTL이라는 수명 값(Lifetime)을 가지고 있고 이 값이 0이 되면 네트워크 장비에서 버려짐
      - ◆ TTL은 실제 초와 같은 시간이 아니라 홉을 지칭하며 하나의 홉을 지날 때마다 TTL 값은 1씩 줄어듬



Router/L3 Switch

- ❖ 경로 지정 – 라우팅/스위칭
  - ✓ 라우터가 경로 정보를 얻는 방법
    - ● 방법
      - ◆ 다이렉트 커넥티드
      - ◆ 스태틱 라우팅
      - ◆ 다이나믹 라우팅

Router/L3 Switch

- ❖ 경로 지정 – 라우팅/스위칭
  - ✓ 라우터가 경로 정보를 얻는 방법
    - ● 다이렉트 커넥티드
      - ◆ IP 주소를 입력할 때 사용된 IP 주소와 서브넷 마스크로 해당 IP 주소가 속한 네트워크 주소 정보를 알 수 있음
      - ◆ 라우터나 PC에서는 이 정보로 해당 네트워크에 대한 라우팅 테이블을 자동으 로 만듬
      - ◆ 이 경로 정보를 다이렉트 커넥티드(Direct Connected)라고 부름
      - ◆ 다이렉트 커넥티드로 생성되는 경로 정보는 인터페이스에 IP를 설정하면 자동 생성되는 정보이므로 정보를 강제로 지울 수 없고 해당 네트워크 설정을 삭제하거나 해당 네트워크 인터페이스가 비활성화되어야만 자동으로 사라짐

R1 라우팅 테이블

목적지	넥스트 홉	인터페이스
1.1.1.0/30	connected	Eth 0/0
10.10.10.0/24	connected	Lo 0

R2 라우팅 테이블

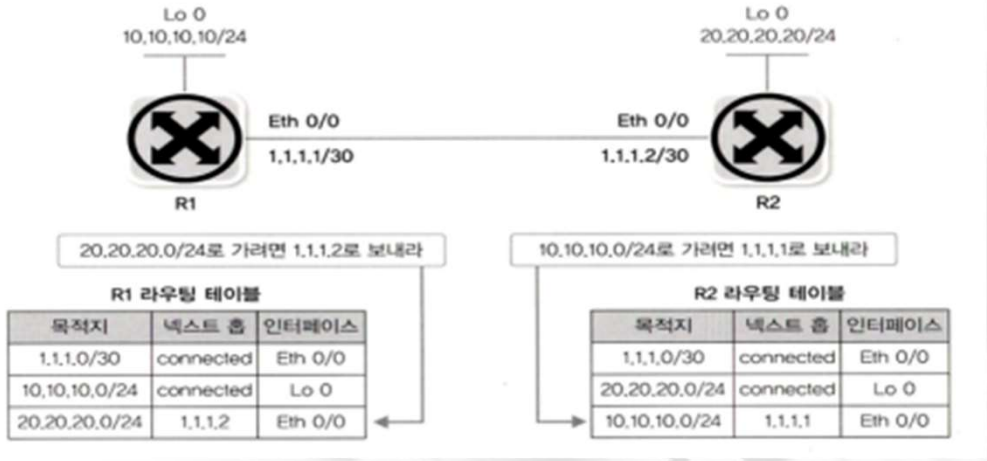
목적지	넥스트 홉	인터페이스
1.1.1.0/30	connected	Eth 0/0
20.20.20.0/24	connected	Lo 0

Router/L3 Switch

- ❖ 경로 지정 – 라우팅/스위칭
  - ✓ 라우터가 경로 정보를 얻는 방법
    - Static Routing
      - ◆ 관리자가 목적지 네트워그와 넥스트 홉을 라우터에 직접 지정해 경로 정보를 입력하는 것이 Static Routing
      - ◆ 스태틱 라우팅은 관리자가 경로를 직접 지정하므로 라우팅 정보를 매우 직관적으로 설정 및 관리
      - ◆ 스태틱 라우팅은 다이렉트 커넥티드처럼 연결된 인터페이스 정보가 삭제되거나 비활성화되면 연관된 스태틱 라우팅 정보가 자동 삭제
      - ◆ 물리 인터페이스가 아닌 논리 인터페이스는 물리 인터페이스가 비활성화되더라도 함께 비활성화되지 않는 경우도 있어 라우팅 테이블에서 사라지지 않을 수 있음

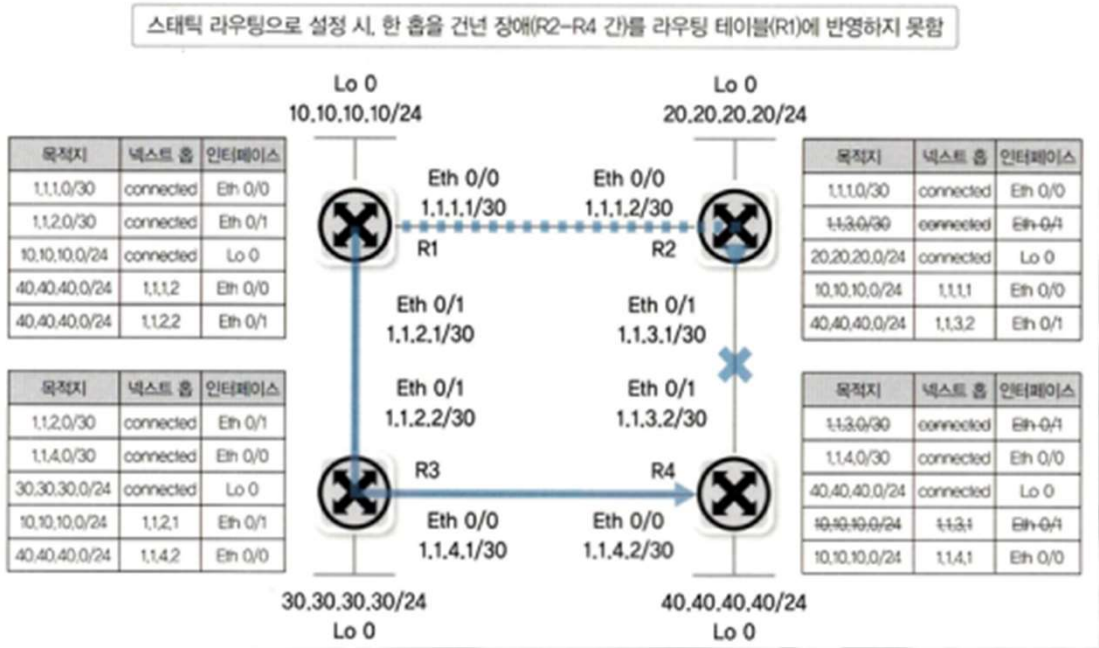
Router/L3 Switch

- ❖ 경로 지정 – 라우팅/스위칭
  - ✓ 라우터가 경로 정보를 얻는 방법
    - Static Routing
      - ◆ R1 은 10.10.10.10/24와 1.1.1.0/30 네트워크만 다이렉트 커넥티드를 이용해 학습할 수 있으므로 20.20.20.20/24 네트워크 정보가 없음
      - ◆ R1 의 10.10.10.0/24 네트워크에서 R2의 20.20.20.20 과 통신하려면 넥스트 홉을 1.1.1.2로 스태틱 라우팅을 설정해야 함
      - ◆ 네트워크 통신은 양방향이므로 되돌아오는 패킷도 고려
        R2에서는 20.20.20.0/24 네트워크 정보와 1.1.1.0/30 네트워크 정보를 다이렉트 커넥티드 라우팅을 이용해 얻을 수 있지만 원래 출발지였던 10.10.10.10/24 네트워크 정보가 없으므로 R2에서도 스태틱 라우팅을 이용해 10.10.10.10/24 네트워크 정보를 추가해야 하는데 넥스트 홉은 1.1.1.1



Router/L3 Switch

- ❖ 경로 지정 – 라우팅/스위칭
  - ✓ 라우터가 경로 정보를 얻는 방법
    - Dynamic Routing
      - ◆ 스태틱 라우팅은 관리자가 변화가 적은 네트워크에서 네트워크를 손쉽게 관리할 수 있는 좋은 방법이지만 큰 네트워크는 스태틱 라우팅만으로는 관리가 어려움
      - ◆ 스태틱 라우팅은 라우터 너머의 다른 라우터의 상태 정보를 파악할 수 없어 라우터 사이의 회선이나 라우터에 장애가 발생하면 장애 상황을 파악하고 대체 경로로 패킷을 보낼 수 없기 때문



목적지	넥스트 홉	인터페이스
1.1.1.0/30	connected	Eth 0/0
1.1.2.0/30	connected	Eth 0/1
10.10.10.0/24	connected	Lo 0
40.40.40.0/24	1.1.1.2	Eth 0/0
40.40.40.0/24	1.1.2.2	Eth 0/1

목적지	넥스트 홉	인터페이스
1.1.1.0/30	connected	Eth 0/0
~~1.1.3.0/30~~	~~connected~~	~~Eth 0/1~~
20.20.20.0/24	connected	Lo 0
10.10.10.0/24	1.1.1.1	Eth 0/0
40.40.40.0/24	1.1.3.2	Eth 0/1

목적지	넥스트 홉	인터페이스
1.1.2.0/30	connected	Eth 0/1
1.1.4.0/30	connected	Eth 0/0
30.30.30.0/24	connected	Lo 0
10.10.10.0/24	1.1.2.1	Eth 0/1
40.40.40.0/24	1.1.4.2	Eth 0/0

목적지	넥스트 홉	인터페이스
~~1.1.3.0/30~~	~~connected~~	~~Eth 0/1~~
1.1.4.0/30	connected	Eth 0/0
40.40.40.0/24	connected	Lo 0
~~10.10.10.0/24~~	~~1.1.3.1~~	~~Eth 0/1~~
10.10.10.0/24	1.1.4.1	Eth 0/0

Router/L3 Switch

- ❖ 경로 지정 – 라우팅/스위칭
  - ✓ 라우터가 경로 정보를 얻는 방법
    - Dynamic Routing
      - ◆ 관리해야 할 네트워크 수가 많아지거나 연결이 복잡해지면 관리자가 직접 관리해야 하는 라우팅 개수가 기하급수적으로 늘어나므로 관리자가 라우팅을 직접 추가하거나 삭제하는 데 한계가 있음
      - ◆ 다이나믹 라우팅(Dynamic Routing)은 스태틱 라우팅의 이런 단점을 보완
      - ◆ 라우터끼리 자신이 알고 있는 경로 정보나 링크 상태 정보를 교환해 전체 네트워크 정보를 학습
      - ◆ 주기적으로 또는 상태 정보가 변경될 때 라우터끼리 경로 정보가 교환되므로 라우터를 연결하는 회선이나 라우터 자체에 장애가 발생하면 이 상황을 인지해 대체 경로로 패킷을 포워딩할 수 있음
      - ◆ 관리자의 개입없이 라우터끼리의 정보교환만으로 장애를 인지하고 트래픽을 우회할 수 있으므로 대부분의 네트워크에서 다이나믹 라우팅을 사용

Router/L3 Switch

- ❖ 경로 지정 – 라우팅/스위칭
  - ✓ 라우터가 경로 정보를 얻는 방법
    - Dynamic Routing
      - ◆ 다이나믹 라우팅에서는 자신이 광고할 네트워크를 선언해주어야 하는데 각 다이나믹 라우팅 프로토콜에 따라 설정 방법만 다를 뿐 광고에 필요한 자신의 네트워크를 선언해야 하는 것은 동일

R1 라우팅 테이블

목적지	넥스트 홉	인터페이스
1.1.1.0/30	connected	Eth 0/0
10.10.10.0/24	connected	Lo 0
20.20.20.0/24	1.1.1.2	Eth 0/0

R2 라우팅 테이블

목적지	넥스트 홉	인터페이스
1.1.1.0/30	connected	Eth 0/0
20.20.20.0/24	connected	Lo 0
10.10.10.0/24	1.1.1.1	Eth 0/0

Router/L3 Switch

- ❖ 경로 지정 – 라우팅/스위칭
  - ✓ 라우터가 경로 정보를 얻는 방법
    - Dynamic Routing
      - ◆ 라우터에서 라우팅의 역할은 경로 정보를 얻는 것뿐만 아니라 다양한 경로 정보를 체계적으로 데이터베이스화하고 순위를 적절히 부여해 최선의 경로 정보만 수집 해두는 것
      - ◆ 패킷을 포워딩할 때 최적의 경로를 찾는 작업을 단순화하기 위해 라우팅 정보를 저장할 때 최적의 경로만 추려 별도 테이블에 미리 보관
      - ◆ 라우터가 수집한 경로 정보, 원시 데이터(Raw Data)를 토폴로지 테이블이라고 하고 이 경로 정보 중 최적의 경로를 저장하는 테이블을 라우팅 테이블이라고 함
      - ◆ 많은 패킷을 실시간으로 포워딩해야 하는 라우터는 라우팅 테이블을 참고해 패킷 경로를 지정하고 포워딩
      - ◆ 패킷을 보낼 때는 전체 경로를 고려하는 것이 아니라 다음 라우터까지만 패킷을 포워딩하는데 이런 기법을 홉 바이 홉 라우팅이라고 하고 다음 라우터까지만 패킷을 보내면 그 다음 라우터가 다시 다음 라우터까지의 최적의 경로로 패킷을 포워딩

Router/L3 Switch

- ❖ 경로 지정 – 라우팅/스위칭
  - ✓ 라우터가 경로 정보를 지정하는 방법
    - 패킷이 들어와 라우팅 테이블을 참조하고 최적의 경로를 찾아 라우터 외부로 포워딩하는 작업을 스위칭이라고 함
    - 이 스위칭은 2계층의 스위치와 이름은 비슷하지만 다른 3계층 장비인 라우터가 패킷 경로를 지정해 보내는 작업을 의미
    - 패킷을 최선의 경로로 내보낼 때도 여러 가지를 고려해야 하는데 들어온 패킷의 목적지가 라우팅 테이블에 있는 정보와 완벽히 일치할 때도 있지만 비슷하게 일치하거나 일치하지 않는 경우도 생길 수 있기 때문

Router/L3 Switch

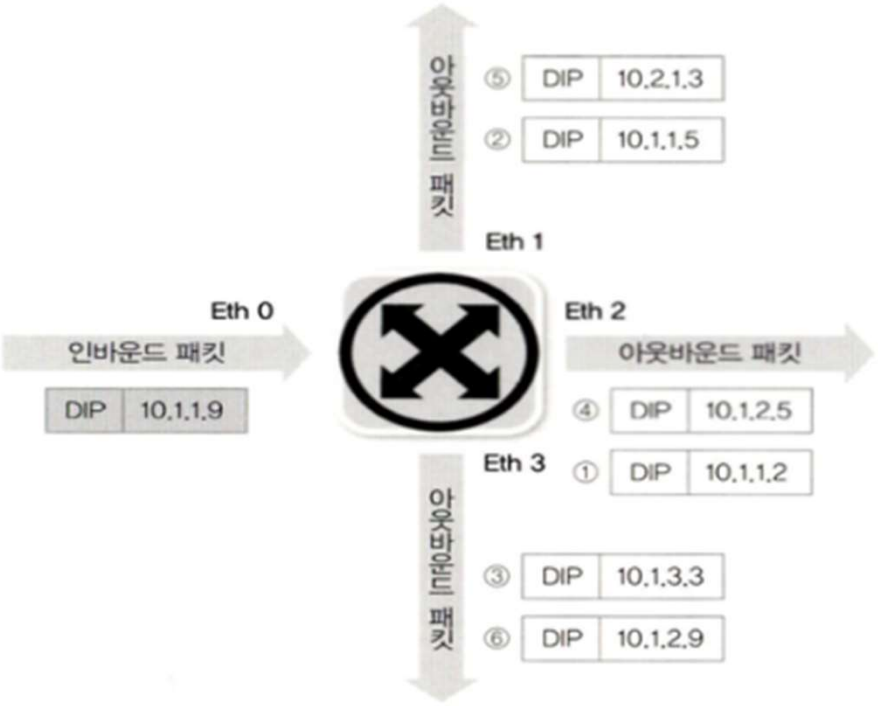
❖ 경로 지정 – 라우팅/스위칭

✓ 라우터가 경로 정보를 지정하는 방법

- 10.1.1.9 IP가 목적지인 패킷이 라우터로 들어온 경우 라우터는 도착지 IP와 가장 가깝게 매치되는 경로 정보를 찾음
- 10.1.1.9와 완전히 매치되는 경로 정보가 없기 때문에 롱기스트 프리픽스 매치(Longest Prefix Match) 기법을 이용해 갖고 있는 경로 정보 중 가장 가까운 경로를 선택

라우팅 테이블

목적지	인터페이스
10.1.1.0/24	Eth 2 ①
10.1.1.5/32	Eth 1 ②
10.1.0.0/16	Eth 3 ③
10.1.2.0/24	Eth 2 ④
10.0.0.0/8	Eth 1 ⑤
10.1.2.9/32	Eth 3 ⑥



Router/L3 Switch

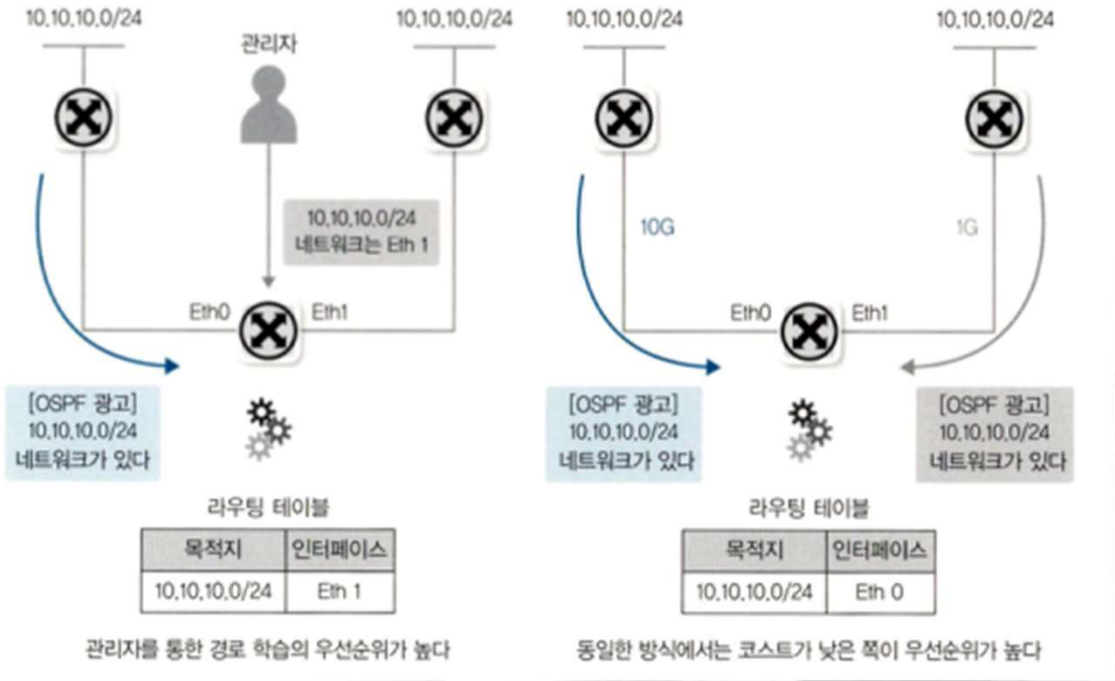
- 경로 지정 – 라우팅/스위칭
  - ✓ 라우터가 경로 정보를 지정하는 방법
    - 라우팅 테이블과 도착지 정보가 매치되는 정보는 10.0.0.0/8, 10.1.0.0/16, 10.1.1.0/24
    - 10.1.2.0/24와 10.1.2.9/32는 세 번째 자리부터 매치되지 않으므로 도착지와 매치되는 정보로 볼 수 없음
    - 마지막 옥텟 정보가 다르므로 10.1.1.5/32 라우팅 정보도 도착지와 매치되는 정보로 볼 수 없음
    - 10.0 .0.0/8, 10.1.0.0/16, 10.1.1.0/24 세 개의 라우팅 정보 중 목적지와 더 많이 매치되는 정보는 10.1.1.0/24
    - 앞의 두 라우팅 정보보다 더 많은 네트워크 정보가 목적지와 매치되므로 이 정보를 최선의 정보로 인식해 Eth 2 인터페이스 쪽으로 패킷을 내보냄
    - 사실 라우터에서 이 작업은 많은 부하가 걸리는데 정확한 정보를 매치하는 이그잭트 매치는 단순한 서치 작업으로 찾고 패킷을 처리할 수 있지만 롱기스트 매치처럼 부정확한 정보 중 가장 비슷한 경로를 찾는 작업은 더 많은 리소스를 소모
    - 라우터에서 패킷이 들어올 때 마다 이 작업을 수행하면 많은 리소스를 소모

Router/L3 Switch

- ❖ 경로 지정 – 라우팅/스위칭
  - ✓ 라우터가 경로 정보를 지정하는 방법
    - 대부분의 라우터는 오래 걸리는 이런 반복 작업을 줄여주는 기술을 채용
    - 한 번 스위칭 작업을 수행한 정보는 캐시에 저장하고 뒤에 들어오는 패킷은 라우팅 테이블을 확인하는 것이 아니라 캐시를 먼저 확인하는 방법인데 이런 기술이 유용한 것은 패킷 네트워크에서 데이터를 보내기 위해 동일한 출발지 IP, 동일한목적지 IP, 포트 번호로 여러 개의 패킷이 연속적으로 보내지기 때문
    - 이런 캐시 기술도 캐시하는 정보에 따라 다양
    - 단순히 목적지 IP만 캐시하는 경우와 출발지와 목적지 IP 모두 캐시하는 경우 및 포트 번호 정보까지 포함해 플로를 모두 저장하는 경우로 구분할 수 있음
    - 한 단계 나아가 넥스트 홉 L2 정보까지 캐시해 스위칭 시간을 줄이는 기법도 사용
    - 이런 캐시 기법은 레디스(Redis)와 같은 메모리 캐시를 이용해 데이터베이스 부하를 줄이는 기 법과 매우 유사

Router/L3 Switch

- ❖ 경로 지정 – 라우팅/스위칭
  - ✓ 라우팅/스위칭 우선 순위
    - 라우팅 테이블은 가장 좋은 경로 정보만 모아놓은 핵심 정보
    - 일반적인 경로 정보를 모아놓은 토폴로지 테이블에서 좋은 경로 정보의 우선순위는 경로 정보를 받은 방법과 거리를 기준으로 정함



Router/L3 Switch

- ❖ 경로 지정 – 라우팅/스위칭
  - ✓ 라우팅/스위칭 우선 순위
    - 목적지 네트워크 정보가 동일한 서브넷을 사용하는 경우 정보를 얻은 소스에 따라 가중치를 정함
    - 이 가중치 값은 라우팅 정보의 분류와 마찬가지로 크게 3가지
      - ◆ 다이렉트 커넥티드
      - ◆ 스태틱 라우팅
      - ◆ 다이나믹 라우팅
    - 3가지 경로 수집 방법 중 우선순위가 가장 높은 것은 라우터에 바로 연결된 네트워크
    - 다이렉트 커넥티드라고 불리는 이 네트워크 정보는 라우터에 바로 붙은 대역이어서 경로 선정 시 우선 순위가 가장 높고 다음은 관리자가 목적지 네트워크에 대한 경로를 직접 지정하는 스태틱 라우팅인데 비록 로컬 네트워크는 아니지만 목적지 네트워크에 대한 경로 정보를 관리자가 직접 입력하면 신뢰도가 높아 로컬 네트워크 다음으로 우선순위가 높음
    - 마지막으로 라우팅 프로토콜로부터 경로를 전달받은 네트워크인데 이 경우는 로컬이나 관리자가 지정한 스태틱 라우팅처럼 경로 정보를 직접 얻은 것이 아니라 다른 라우터를 통해 경로를 전달받았기 때문에 우선순위가 낮음
    - 라우팅 프로토콜 중에서도 어떤 라우팅 프로토콜을 통해 경로 정보를 얻었는가에 따라 우선순위가 다름

Router/L3 Switch

- ❖ 경로 지정 – 라우팅/스위칭
  - ✓ 라우팅/스위칭 우선 순위
    - 기본적인 우선순위는 미리 정해져 있지만 필요에 따라 관리자가 우선순위를 조정해 라우팅 경로를 조정할 수 있음
    - 이런 우선순위를 AD(Administrative Distance - 관리 거리)라고 부르며 라우터 생산업체마다 값이 조금씩 다름
    - Cisco 장비 기준

0	Connected Interface(다이렉트 커넥티드)
1	Static Route(스태틱 라우팅)
20	External BGP
110	OSPF
115	IS-IS
120	RIP
200	Internal BGP
255	Unknown

Router/L3 Switch

- ❖ 경로 지정 – 라우팅/스위칭
  - ✓ 라우팅/스위칭 우선 순위
    - 경로 정보를 얻은 소스가 같아 가중치 값이 동일한 경우에는 코스트(Cost) 값으로 우선 순위를 정함
    - 코스트 값까지 동일한 경우에는 ECMP(Equal-Cost Multi-Path) 기능으로 동일한 코스트 값을 가진 경로 값 정보를 모두 활용해 트래픽을 분산
    - 코스트 값은 일종의 거리를 나타내는 값으로 라우팅 프로토콜마다 기준이 다름
    - RIP은 홉 수, OSPF는 대역폭(Bandwidth), EIGRP는 다양한 값들을 연산해 나온 값으로 코스트를 지정
    - 라우터의 라우팅, 스위칭 우선 순위

우선순위	구분	적용 방법
1	롱기스트 매치	스위칭
2	AD(관리 거리)	라우팅
3	코스트	라우팅
4	부하 분산(ECMP)	라우팅

4계층 장비

❖ 개요

- ✓ 기존 네트워크 장비는 스위치와 라우터처럼 2계층이나 3계층에서 동작하는 장비를 지칭하는 용어였지만 IP 부족으로 NAT 기술이 등장하고 보안용 방화벽, 프록시와 같은 장비들이 등장하면서 4계층 이상에서 동작하는 네트워크 장비가 많아지면서 4계층에서 동작하는 장비도 네트워크 장비에 포함
- ✓ 4계층의 특징인 포트 번호, 시퀀스 번호, ACK 번호를 붙여서 사용
- ✓ 기존 2. 3계층 장비에서 고려하지 않았던 통신의 방향성이나 순서와 같은 통신 전반에 대한 관리가 필요하며 이런 정보를 Session Table 이라는 공간에 담아 관리
- ✓ 4계층 이상에서 동작하는 네트워크 장비는 이런 세션 테이블을 관리해야 하고 이 정보를 기반으로 동작

4계층 장비

❖ 특징

- ✓ 4계층 장비는 TCP와 같은 4계층 헤더에 있는 정보를 이해하고 이 정보들을 기반으로 동작
- ✓ 4계층 이상에서 동작하는 Load Balancer, 방화벽과 같은 장비를 세션 장비라고 부르기도 함
- ✓ 세션 장비는 추가로 고려해야 할 특징이 많은데 그 중 최우선적으로 고려할 요소
  - 세션 테이블
    - ◆ 세션 장비는 세션 테이블 기반으로 운영
    - ◆ 세션 정보를 저장 및 확인하는 작업 전반에 대한 이해가 필요
    - ◆ 세션 정보는 세션 테이블에 남아 있는 라이프 타임이 존재하므로 이 부분에 대한 고려가 필요
  - Symmetric(대칭) 경로 요구: Inbound와 Outbound 경로가 일치해야 함
  - 정보 변경(Load Balancer의 경우): IP 주소가 변경되며 확장된 L7 Load Balancer(ADC)는 애플리케이션 프로토콜 정보도 변경됨
- ✓ 세션 장비의 이런 요소가 서비스에 영향을 미치므로 네트워크 통신 중간 위치에 세션을 기반으로 동작하는 방화벽, NAT, Load Balancer와 같은 장비가 있을 경우 네트워크 인프라뿐만 아니라 시스템 설계와 애플리케이션 개발에도 세션 장비에 대한 고려가 필요

4계층 장비

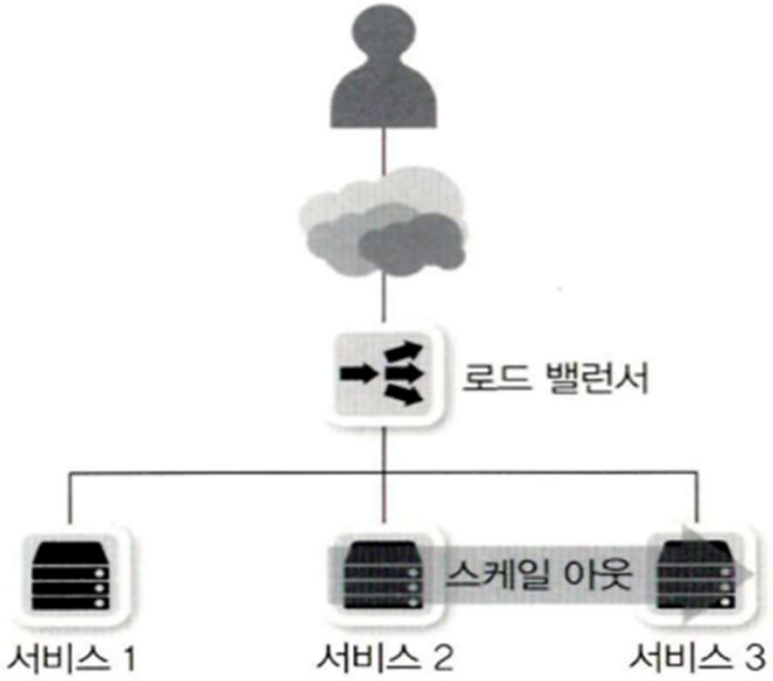
❖ Load Balancer

✓ 개요

- 서버나 장비의 부하를 분산하기 위해 사용하는 장비가 Load Balancer
- 4계층 이상에서 동작하면서 IP 주소나 4계층 정보 또는 애플리케이션 정보를 확인 및 수정하는 기능도 있지만 가장 많이 쓰이는 분야는 서버의 부하 분산
- 사용자 천 명의 요청을 동시에 처리해주는 서버보다 사용자 5천 명의 요청을 동시에 처리해주는 서버의 비용은 5배가 아니라 이보다 훨씬 많이 소모되는데 내부 부품을 이중화하거나 용량이 더 큰 부품을 사용하면 비용이 크게 올라가므로 작은 장비 여러 대를 묶어 사용하는 방법을 선호하는데 이런 시스템 확장 방법이 스케일 아웃
- 작은 시스템 여러 대를 운영하더라도 사용자는 서버 배치와 상관없이 하나의 서비스로 보여야 하는데 Load Balancer가 서비스에 사용되는 대표 IP 주소를 서비스 IP로 갖고 그 밑에 시스템이 늘어나면 Load Balancer가 각 시스템의 실제 IP로 변경해 요청을 전송
- Load Balancer는 웹 애플리케이션뿐만 아니라 FWLB(FireWall Load Balancing), VPNLB(VPN Load Balancing) 와 같이 다양한 서비스를 위해 사용될 수 있음

4계층 장비

- ❖ Load Balancer
  - ✓ 개요



4계층 장비

- ❖ Load Balancer
  - ✓ L4 Load Balancing & L7 LoadB alancing
    - L4 Load Balancing
      - ◆ 일반적인 Load Balancer가 동작하는 방식
      - ◆ TCP, UDP 정보(포트 넘버)를 기반으로 Load Balancing을 수행
      - ◆ 최근 Load Balancer는 L4, L7의 기능을 모두 지원하므로 L4 Load Balancing만 제공하는 전용 장비는 찾기 힘들지만 장비에서 L7 지원 여부와 상관없이 4계층에 대한 정보로만 분산 처리하는 경우를 L4 Load Balancing이라고 함
    - L7 Load Balancing
      - ◆ HTTP, FTP, SMTP와 같은 애플리케이션 프로토콜 정보를 기반으로 Load Balancing을 수행
      - ◆ HTTP 헤더 정보나 URI와 같은 정보를 기반으로 프로토콜을 이해한 후 부하를 분산할 수 있음
      - ◆ 일반적으로 이런 장비를 ADC(Application Delivery Controller)라고 부르며 프록시(Proxy) 역할을 수행
      - ◆ Squid 나 Nginx에서 수행하는 리버스 프록시(Reverse Proxy) 와 유사한 기능

4계층 장비

- ❖ Load Balancer
  - ✓ L4 Load Balancing & L7 Load Balancing
    - ● 데이터 센터에서 사용하는 Load Balancing 장비는 L4, L7을 모두 지원하며 실제로 어떻게 설정했는가에 따라 L4 Load Balancing과 L7 Load Balancing으로 나누는데 클라우드에서는 L4 Load Balancing과 L7 Load Balancing을 지원하는 컴포넌트를 계층별로 구분해 전용으로 사용하는데 AWS의 NLB(Network Load Balancer)가 L4 Load Balancing ALB(Application Load Balancer)가 L7 Load Balancing용 전용 컴포넌트

4계층 장비

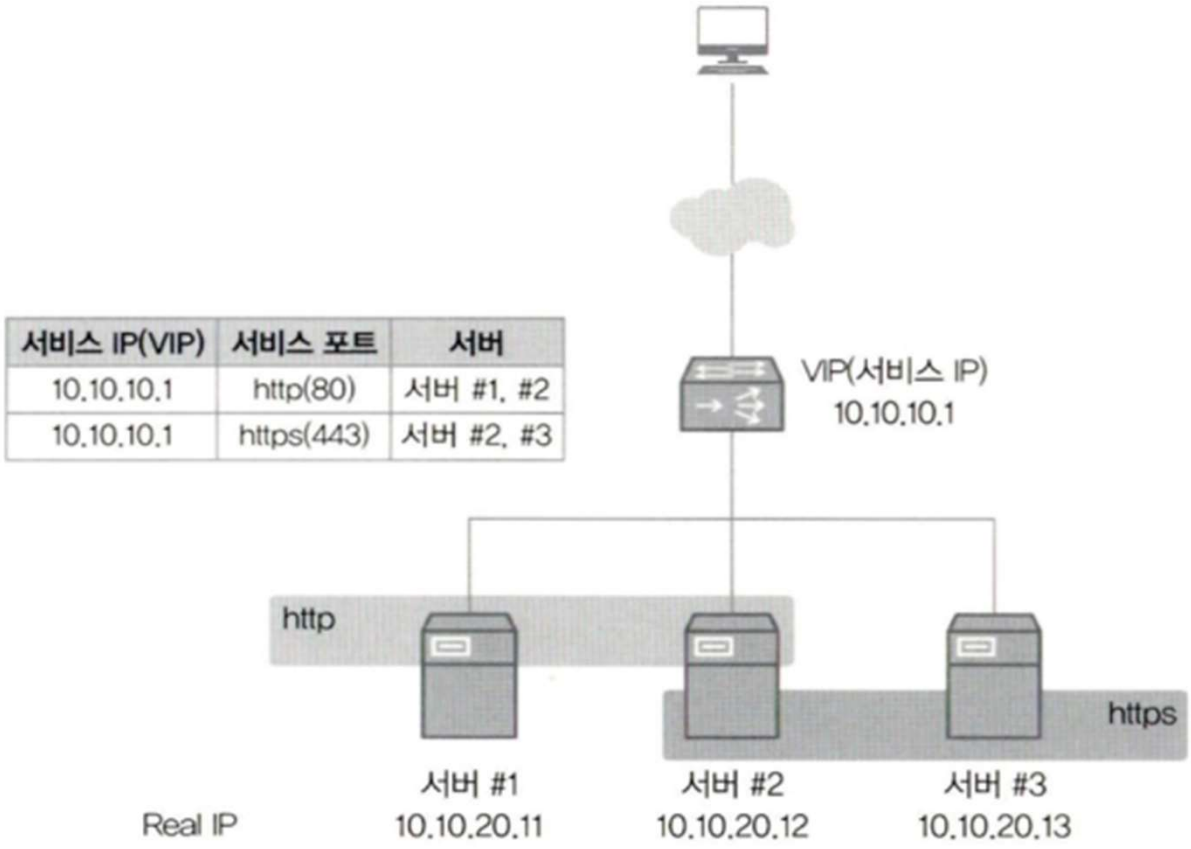
- ❖ Load Balancer
  - ✓ L4 Switch
    - ● 특징
      - ◆ L4 스위치는 용어 그대로 4계층에서 동작하면서 Load Balancer 기능이 있는 스위치
      - ◆ 내부 동작 방식은 4계층 Load Balancer이지만 외형은 스위치처럼 여러 개의 포트를 가지고 있음
      - ◆ 서버형 Load Balancer나 소프트웨어 형태의 Load Balancer도 있지만 다양한 네트워크 구성이 가능한 스위치형 Load Balancer가 가장 대중화되어 있음
      - ◆ L4 스위치는 부하 분산 및 성능 최적화 와 리다이렉션 기능을 제공

4계층 장비

- ❖ Load Balancer
  - ✓ L4 Switch
    - ● L4 스위치를 이용한 부하 분산
      - ◆ L4 스위치가 동작하려면 Virtual Server, Virtual IP, Real Server 와 Real IP를 설정해야 함
      - ◆ 가상 서버는 사용자가 바라보는 실제 서비스이고 가상 IP는 사용자가 접근해야 하는 서비스 IP 주소
      - ◆ Real Server는 실제 서비스를 수행하는 서버이고 Real IP는 실제 서버의 IP
      - ◆ L4 스위치는 가상 IP를 Real IP로 변경해주는 역할을 수행
      - ◆ 사용자는 L4 스위치의 가상 IP를 목적지로 서비스를 요청하고 L4 스위치가 목적지로 설정된 가상IP를 Real IP로 다시 변경해 전송
      - ◆ 이 과정에서 부하를 어떤 방식으로 분산할지 결정할 수 있음

4계층 장비

- ❖ Load Balancer
  - ✓ L4 Switch
    - ● L4 스위치를 이용한 부하 분산

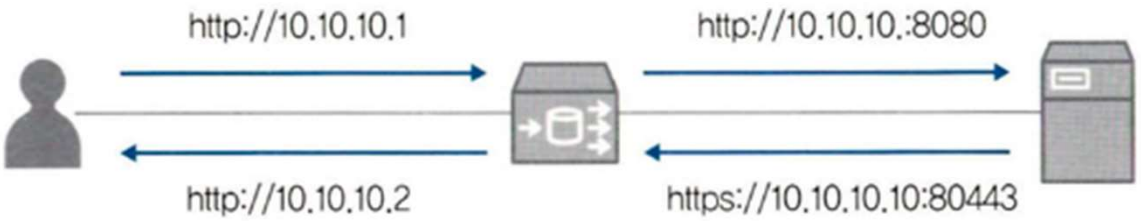


4계층 장비

- ❖ Load Balancer
  - ✓ L4 Switch
    - ● L4 스위치를 이용한 부하 분산
      - ◆ 현재 서버 세 대가 있는데 서버의 각 IP 주소는 10.10.20.11. 10.10.20.12, 10.10.20.13
      - ◆ 1번은 http 3번은 https 서비스 Daemon이 동작하고 서버 2번만 http와 https 서비스 Daemon이 모두 동작
      - ◆ 사용자가 http 와 https 서비스로 접근하기 위한 VIP 주소인 10.10.10.1이 Load Balancer에 설정되어 있음
      - ◆ VIP에는 사용자의 서비스 요청이 들어올 때 어느 서버로 요청을 전달할 것인지 부하 분산 그룹을 설정
      - ◆ http 서비스는 서버 1, 2번과 https 서비스는 서버 2번과 3번으로 부하 분산 그룹이 있음
      - ◆ Load Balancer에서 부하 분산을 위한 그룹을 만들 때는 OSI 3계층 정보인 IP 주소 뿐 아니라 4계층 정보인 서비스 포트까지 지정해 생성
      - ◆ Load Balancer를 L4 스위치 또는 7계층 정보까지 확인해 처리하는 기능이 포함되는 경우도 있어 L7 스위치라고도 하지만 보통 Load Balancer를 L4 스위치 라고 부름

4계층 장비

- ❖ Load Balancer
  - ✓ L4 Switch
    - ● L4 스위치를 이용한 부하 분산
      - ◆ HTTP와 HTTPS 서비스에 대해 각각 동일한 VIP를 사용했지만 서로 다른 VIP로도 구성할 수 있으며 Load Balancer의 VIP에 설정된 서비스 포트와 실제 서버의 서비스 포트는 반드시 같을 필요가 없음
      - ◆ 실제 서버에서는 서비스 포트 8080을 이용해 웹 서비스를 수행하면서 VIP에서는 일반 HTTP 서비스 포트인 80번 포트로 설정할 수 있는데 이렇게 되면 사용자는 VIP의 80 서비스 포트로 접근하고 Load Balancer에서는 해당 서비스 요청을 실제 서버 8080 서비스 포트로 포트 변경까지 함께 수행



VIP	포트
10.10.10.1	80
10.10.10.2	443

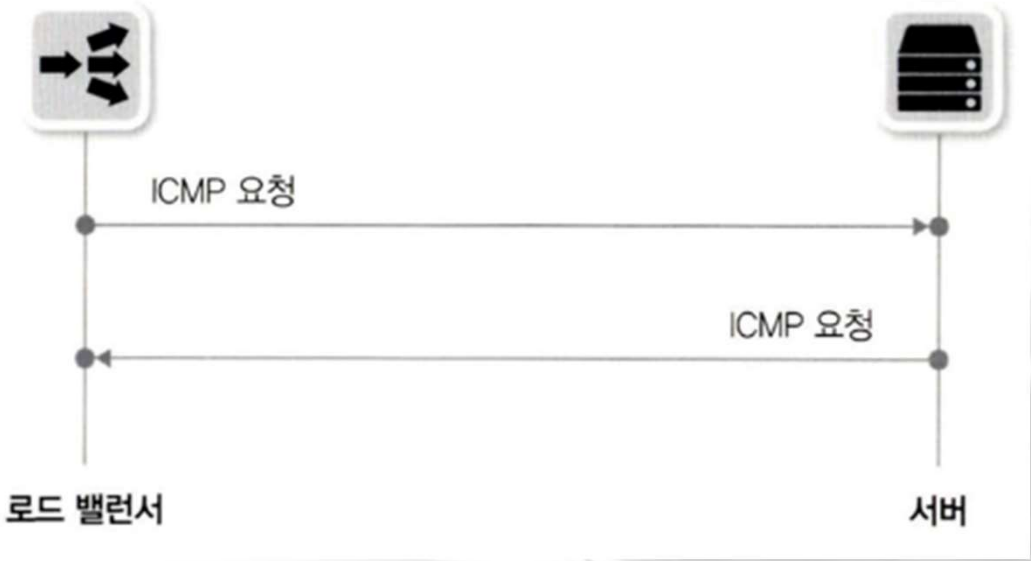
리얼 IP	포트
10.10.20.10	8080
10.10.20.10	80443

4계층 장비

- ❖ Load Balancer
  - ✓ 헬스 체크
    - ● 개요
      - ◆ Load Balancer에서는 부하 분산을 하는 각 서버의 서비스를 주기적으로 헬스 체크(Health Cheek)해 정상적인 서비스 쪽으로만 부하를 분산하고 비정상적인 서버는 서비스 그룹에서 제외해 트래픽을 보내지 않음
      - ◆ 서비스 그룹에서 제외된 후에도 헬스 체크를 계속 수행해 다시 정상으로 확인되면 서비스 그룹에 해당 장비를 다시 넣어 트래픽이 서버 쪽으로 보내지도록 함

4계층 장비

- ❖ Load Balancer
  - ✓ 헬스 체크
    - ● 헬스 체크 방식
      - ◆ ICMP
        - ▪ VIP에 연결된 Real Server에 대해 ICMP(ping)로 헬스 체크를 수행하는 방법
        - ▪ 살아 있는지 여부만 체크하는 방법이므로 잘 사용하지 않음



4계층 장비

- ❖ Load Balancer
  - ✓ 헬스 체크
    - ● 헬스 체크 방식
      - ◆ TCP 서비스 포트
        - ▪ 가장 기본적인 헬스 체크 방법은 Load Balancer에 설정된 서버의 서비스 포트를 확인하는 것
        - ▪ Load Balancer에서 서버의 서비스 포트 2000번을 등록했다면 Load Balancer에서는 Real IP의 2000번 포트로 SYN을 보내고 해당 Real IP를 가진 서버로부터 SYN, ACK를 받으면 서버에 다시 ACK로 응답하고 FIN을 보내 헬스 체크를 종료
        - ▪ 서비스 포트를 이용해 헬스 체크를 할 때는 실제 서비스 포트가 아닌 다른 서비스 포트로도 가능

4계층 장비

- ❖ Load Balancer
  - ✓ 헬스 체크
    - ● 헬스 체크 방식
      - ◆ TCP 서비스 포트: Half Open
        - ▪ 일반 TCP 서비스 포트를 확인할 때는 SYN/SYN, ACK/ACK까지 정상적인 3방향 핸드셰이크를 거치게 됨
        - ▪ 헬스 체크로 인한 부하를 줄이거나 정상적인 종료 방식보다 빨리 헬스 체크 세션을 끊기 위해 정상적인 3방향 핸드셰이크와 4방향 핸드셰이크가 아닌 TCP Half Open(절반 개방) 방식을 사용하기도 함
        - ▪ TCP Half Open 방식은 초기의 3방향 핸드셰이크와 동일하게 SYN을 보내고 SYN, ACK를 받지만 이후 ACK 대신 RST를 보내 세션을 끊음

4계층 장비

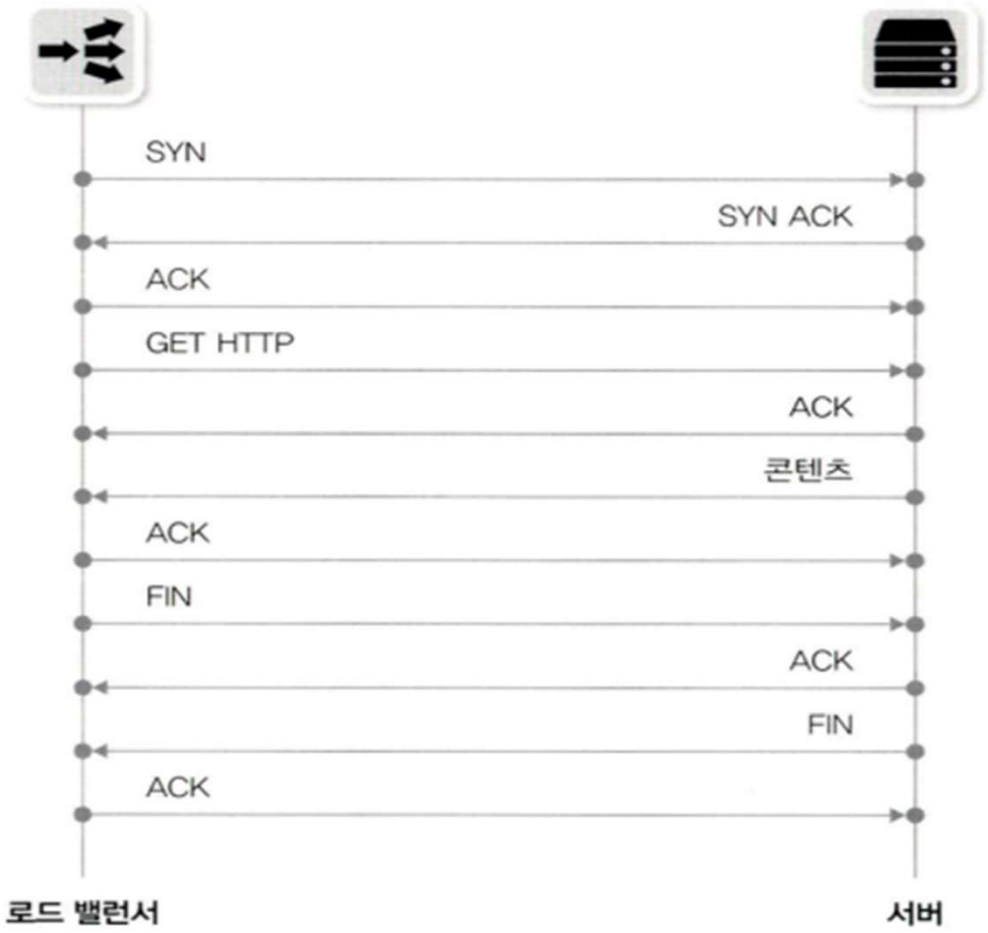
- ❖ Load Balancer
  - ✓ 헬스 체크
    - ● 헬스 체크 방식
      - ◆ HTTP 상태 코드
        - ▪ 웹 서비스를 할 때 서비스 포트까지는 TCP로 정상적으로 열리지만 웹 서비스에 대한 응답을 정상적으로 해주지 못하는 경우가 있음
        - ▪ Load Balancer의 헬스 체크 방식 중 HTTP 상태 코드를 확인하는 방식으로 Load Balancer가 서버로 3방향 핸드셰이크를 거치고 나서 HTTP를 요청해 정상적인 상태 코드(200 OK)를 응답하는지 여부를 체크해 헬스 체크를 수행할 수 있음

4계층 장비

- ❖ Load Balancer
  - ✓ 헬스 체크
    - ● 헬스 체크 방식
      - ◆ 콘텐츠 확인
        - ▪ Load Balancer에서 서버로 콘텐츠를 요청하고 응답 받은 내용을 확인하여 지정된 콘텐츠가 정상적으로 응답 했는지를 확인하는 방식
        - ▪ 특정 웹 페이지를 호출해 사전에 지정한 문자열이 해당 웹 페이지 내에 포함되어 있는지를 체크하는 것
        - ▪ 이 헬스 체크 방식을 사용하면 Load Balancer에서 직접 관리하는 서버의 상태 뿐 아니라 해당 서버의 Back End(Real Server가 웹 서버인 경우 WAS 서버나 데이터베이스가 Back End)의 상태를 해당 웹 페이지로 체크할 수 있는데 앞 단의 서버가 Back End로 요청을 하고 Back End에서 정상적인 결과 값으로 웹 페이지에 특정 문자열을 출력하게 해 Back End 상태까지 확인하면서 헬스 체크를 수행하는 것
        - ▪ 한 가지 유의사항은 단순히 서버에서 응답 받은 문자열만 체크하면 정상적인 요청 결과 값이 아닌 문자열만 체크하므로 비정상적인 에러 코드에 대한 응답인 경우라도 해당 응답 내용에 헬스 체크를 하려고 했던 문자열이 포함되어 있으면 헬스 체크를 정상으로 판단할 수 있다는 것
        - ▪ 문자열을 이용한 헬스 체크를 수행할 때는 정상 코드 값도 중복으로 확인하거나 문자열 자체를 일반적이 아닌 특정 문자열로 지정해 결과가 정상일 때만 헬스 체크가 성공할 수 있도록 해야 함

4계층 장비

- ❖ Load Balancer
  - ✓ 헬스 체크
    - ● 헬스 체크 방식
      - ◆ 콘텐츠 확인

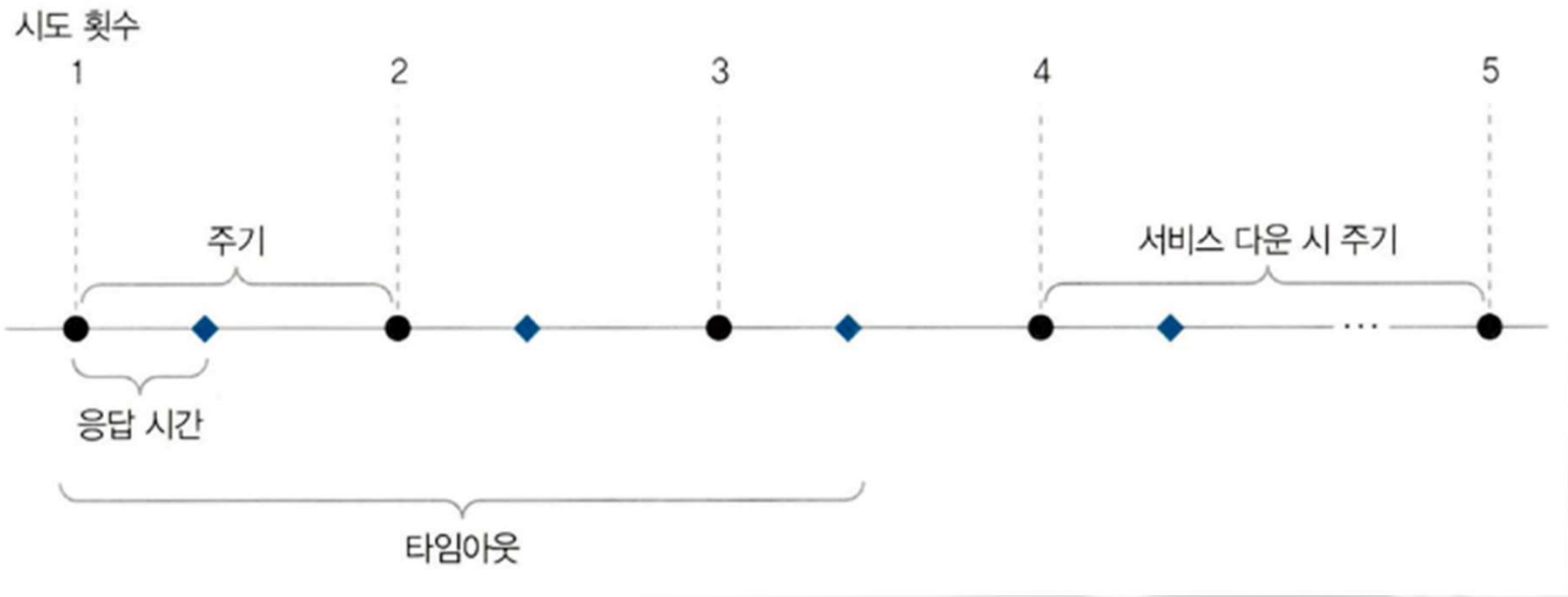


4계층 장비

- ❖ Load Balancer
  - ✓ 헬스 체크
    - ● 헬스 체크 주기와 타이머
      - ◆ 헬스 체크 주기 와 관련된 타이머 값
        - ▪ 주기(Interval): Load Balancer에서 서버로 헬스 체크 패킷을 보내는 주기
        - ▪ 응답 시간(Response): Load Balancer에서 서버로 헬스 체크 패킷을 보내고 응답을 기다리는 시간으로 해당 시간까지 응답이 오지 않으면 실패로 간주
        - ▪ 시도 횟수(Retries): Load Balancer에서 헬스 체크 실패 시 최대 시도 횟수로 최대 시도 횟수 이전에 성공 시 시도 횟수는 초기화
        - ▪ 타임아웃(Timeout): Load Balancer에서 헬스 체크 실패 시 최대 대기 시간으로 헬스 체크 패킷을 서버로 전송한 후 이 시간 내에 성공하지 못하면 해당 서버는 다운
        - ▪ 서비스 다운 시의 주기(Dead Interval): 서비스의 기본적인 헬스 체크 주기가 아닌 서비스 다운 시의 헬스 체크 주기로 서비스가 죽은 상태에서 헬스 체크 주기를 별도로 더 늘릴 때 사용

4계층 장비

- ❖ Load Balancer
  - ✓ 헬스 체크
    - ● 헬스 체크 주기와 타이머



4계층 장비

- ❖ Load Balancer
  - ✓ 헬스 체크
    - ● 헬스 체크 주기와 타이머
      - ◆ 원 모양은 Load Balancer에서 서버로 헬스 체크 패킷을 보내는 시점을 나타냄
      - ◆ 헬스 체크 주기 시간마다 Load Balancer는 서버로 헬스 체크 패킷을 전송
      - ◆ 주기가 3초로 설정되었다면 3초 마다 헬스 체크 패킷을 서버로 전송하는데 이때 원 모양 사이의 시간이 3초
      - ◆ 마름모 모양은 Load Balancer가 보낸 헬스 체크 패킷에 대한 서버의 응답을 최대로 기다리는 시간으로 응답 시간으로 설정된 시간 내에 서버에서 응답이 오지 않으면 Load Balancer는 해당 헬스 체크 시도를 실패로 처리
      - ◆ 응답 시간을 1초로 주었다면 원 모양의 헬스 체크 패킷 전송 이후 1초 내에 서버에서 응답을 수신해야 하는데 이때 유의사항은 헬스 체크 패킷을 보내는 주기를 응답 시간보다 크게 설정해야 한다는 것
      - ◆ Load Balancer가 서버에서 정상적인 응답을 받지 못하면 Load Balancer는 정해진 시도 횟수만큼 헬스 체크를 다시 시도하고 이후에도 서버로부터 응답을 받지 못하면 해당 서비스는 다운된 것으로 체크되어 해당 서비스가 동작하는 서버로 트래픽이 분산되지 않음

4계층 장비

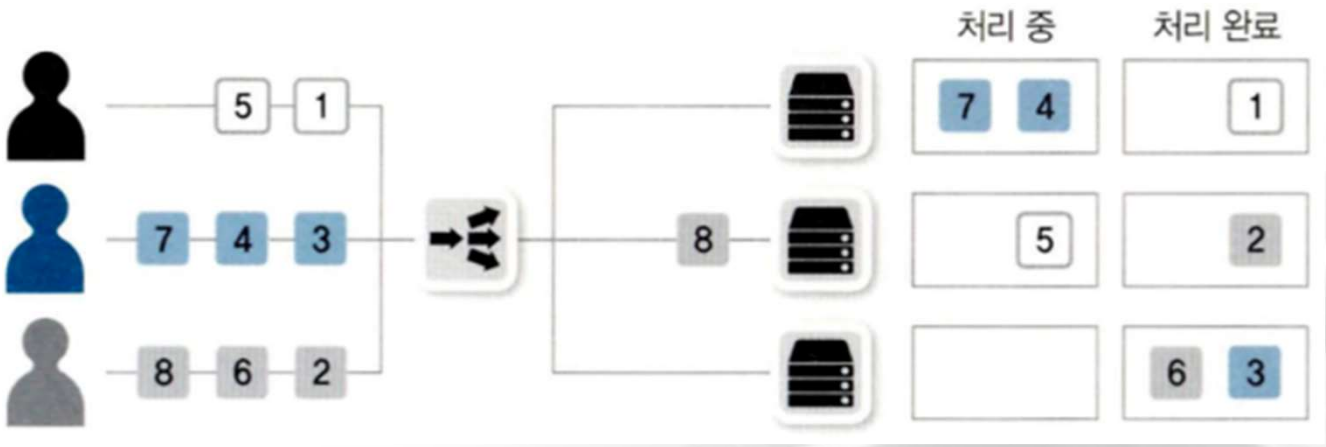
- ❖ Load Balancer
  - ✓ 헬스 체크
    - ● 헬스 체크 주기와 타이머
      - ◆ 서비스 다운까지의 동작을 헬스 체크 주기와 시도 횟수, 응답시간으로 산정하거나 전체 타임아웃 시간으로 산정하기도 함
      - ◆ 헬스 체크가 실패한 첫 번째 시도 시간부터 사전에 정해진 타임아웃까지 헬스 체크가 실패하면 서비스 다운으로 체크할 수도 있음
      - ◆ 서비스 다운까지 시간이 동일한 경우
        - ▪ 주기 3초 시도 횟수 2회 응답시간 1초: (3초 * 3회) + 1초 = 10초
        - ▪ 주기 3초 타임아웃 10초
      - ◆ 서비스가 다운된 후에는 기본 헬스 체크 주기마다 헬스 체크를 수행하지 않고 더 긴 주기로 헬스 체크를 수행하는 기능이 있는 Load Balancer도 있음
      - ◆ 서비스가 다운된 시점에 헬스 체크 주기 시간을 길게 늘여 부하를 감소시킬 수 있다는 장점이 있지만 서비스가 다시 올라오는 시간이 늦어진다는 단점도 있음

4계층 장비

- ❖ Load Balancer
  - ✓ 부하 분산 알고리즘
    - ● 종류
      - ◆ Round Robin
      - ◆ Least Connection
      - ◆ Weighted Round Robin
      - ◆ Weighted Least Connection
      - ◆ Hash

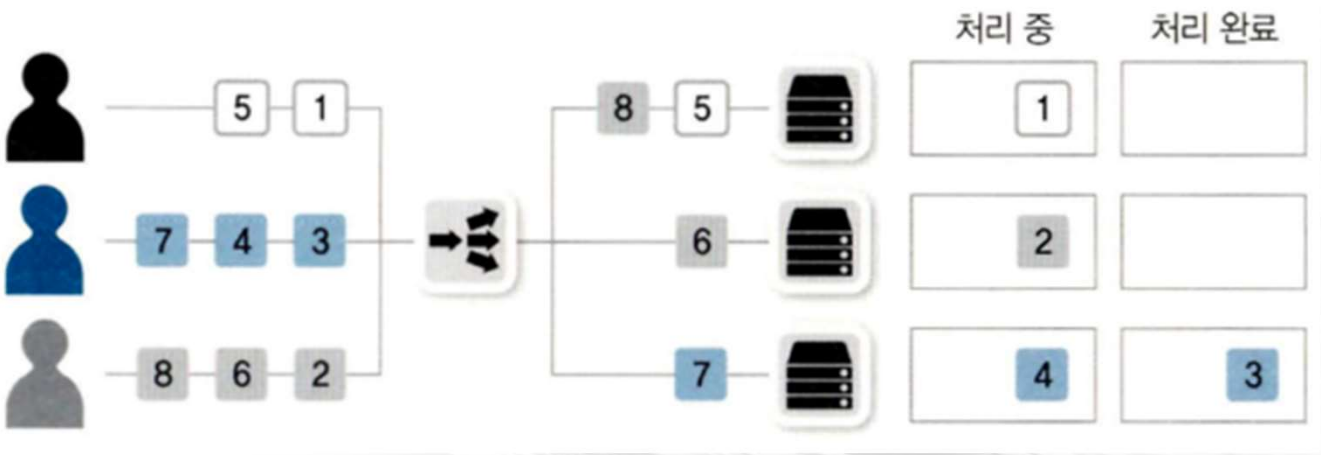
4계층 장비

- ❖ Load Balancer
  - ✓ 부하 분산 알고리즘
    - ● 라운드 로빈
      - ◆ 라운드 로빈 방식은 특별한 규칙 없이 현재 구성된 장비에 순차적으로 돌아가면서 트래픽을 분산
      - ◆ 서버 세 대가 있을 때 첫 번째 요청은 1번 서버 두 번째 요청은 2번 서버 세 번째 요청은 3번 서버 네 번째 요청은 다시 1번 서버로 할당하는데 순차적으로 모든 장비에 분산하므로 모든 장비의 총 누적 세션 수는 동일



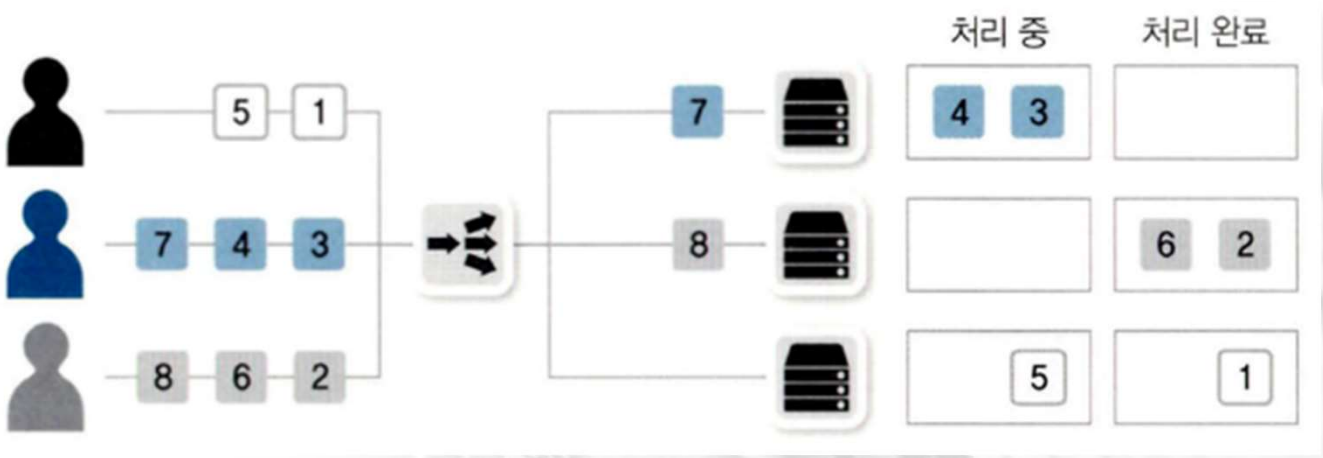
4계층 장비

- ❖ Load Balancer
  - ✓ 부하 분산 알고리즘
    - ● 최소 접속 방식
      - ◆ 최소 접속 방식은 서버가 가진 세션 부하를 확인해 그것에 맞게 부하를 분산하는 Load Balancer에서는 서비스 요청을 각 장비로 보내줄 때마다 세션 테이블이 생성되므로 각 장비에 연결된 현재 세션 수를 알 수 있음
      - ◆ 최소 접속 방식은 각 장비의 세션 수를 확인해 현재 세션이 가장 적게 연결된 장비로 서비스 요청을 보내는 방식
      - ◆ 서비스 별로 세션 수를 관리하면서 분산해주므로 각 장비에서 처리되는 활성화 세션 수가 비슷하게 분산되면서 부하를 분산



4계층 장비

- ❖ Load Balancer
  - ✓ 부하 분산 알고리즘
    - ● 해시
      - ◆ 해시 방식은 서버의 부하를 고려하지 않고 클라이언트가 같은 서버에 지속적으로 접속하도록 하기 위해 사용하는 부하 분산 방식
      - ◆ 서버 상태를 고려하는 것이 아니라 해시 알고리즘을 이용해 얻은 결과 값으로 어떤 장비로 부하를 분산할지를 결정
      - ◆ 알고리즘에 의한 계산 값에 의해 부하를 분산하므로 같은 알고리즘을 사용하면 항상 동일한 결과 값을 가지고 서비스를 분산할 수 있음
      - ◆ 알고리즘 계산에 사용되는 값들을 지정할 수 있는데 주로 출발지 IP 주소, 출발지 서비스 포트, 목적지 서비스 포트를 사용



4계층 장비

- ❖ Load Balancer
  - ✓ 부하 분산 알고리즘
    - ● 해시
      - ◆ 라운드 로빈이나 최소 접속 방식은 부하를 비교적 비슷한 비율로 분산시킬 수 있다는 장점이 있지만 동일한 출발지에서 Load Balancer를 거친 서비스 요청이 처음에 분산된 서버와 그 다음 요청이 분산된 서버가 달라질 수 있어 각 서버에서 세션을 유지해야 하는 서비스는 정상적으로 서비스되지 않는데 이런 문제를 장바구니 문제라고 하는 데 A 서버에 접속해 장바구니에 상품을 넣었는데 두 번째 접속할 때 B 서버로 접속되면 장바구니에 넣은 상품이 보이지 않을 때가 있음
      - ◆ 해시 방식은 알고리즘으로 계산한 값으로 서비스를 분산하므로 항상 동일한 장비로 서비스가 분산되는데 세션을 유지해야 하는 서비스에 적합한 분산 방식
      - ◆ 알고리즘의 결과 값이 특정한 값으로 치우치면 부하 분산 비율이 한쪽으로 치우칠 수도 있는데 최근에는 이런 경우가 별로 없음

4계층 장비

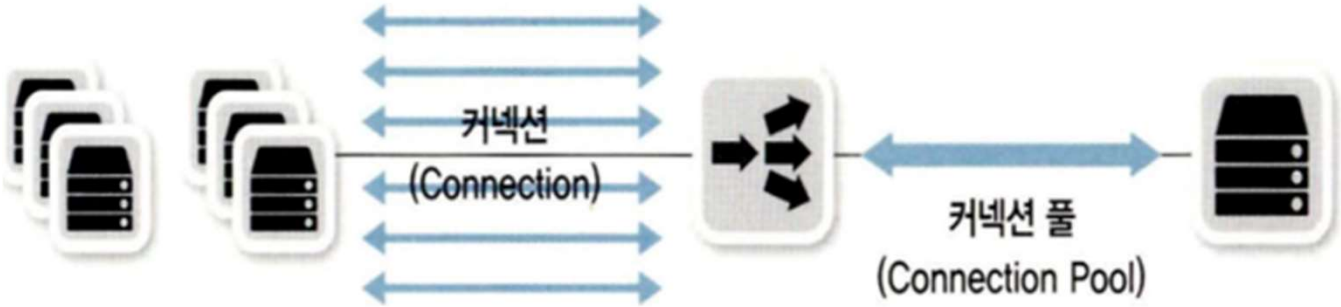
- ❖ Load Balancer
  - ✓ 부하 분산 알고리즘
    - ● 해시
      - ◆ 해시를 사용해야 하는 이유와 최소 접속 방식의 장점을 묶어 부하 분산하는 방법도 있는데 라운드 로빈 방식이나 최소 접속 방식을 사용하면서 Sticky 옵션을 주어 한 번 접속한 커넥션을 지속적으로 유지하는 기법으로 처음 들어온 서비스 요청을 세션 테이블에 두고 같은 요청이 들어오면 같은 장비로 분산해 세션을 유지하는 방법인데 이렇게 하더라도 해당 세션 테이블에는 타임아웃이 있어 타임아웃 이후에는 분산되는 장비가 달라질 수 있다는 것을 고려해야 하며 Sticky 옵션을 사용할 때는 애플리케이션 세션 유지 시간이나 일반 사용자들의 애플리케이션 행동 패턴을 충분히 감안해야 함

4계층 장비

- ❖ Load Balancer
  - ✓ ADC
    - ● 특징
      - ◆ ADC(Application Delivery Controller)는 애플리케이션 계층에서 동작하는 Load Balancer
      - ◆ 4계층에서 동작하는 L4 스위치와 달리 애플리케이션 프로토콜의 헤더와 내용을 이해하고 동작하므로 다양한 부하 분산, 정보 수정, 정보 여과 기능이 가능
      - ◆ ADC는 이런 상세한 동작을 위해 Proxy로 동작
      - ◆ 일부 소프트웨어 ADC를 제외한 대부분의 ADC는 L4 스위치의 기능을 포함하고 있음
      - ◆ 대부분의 ADC는 4계층에서 애플리케이션 계층까지 Load Balancing 기능을 제공하고 Failover(장애 극복 기능), Redirection 기능도 함께 수행하며 이 외에도 애플리케이션 프로토콜을 이해하고 최적화하는 다양한 기능을 제공
      - ◆ Caching, 압축(Compression), 콘텐츠 변환 및 재 작성, Encoding 변환 등이 가능하고 애플리케이션 프로토콜 최적화 기능도 제공
      - ◆ 플러그인 형태로 보안 강화 기능을 추가로 제공해 WAF(Web Application Firewall) 기능이나 HTML, XML 검증과 변환을 수행할 수 있음

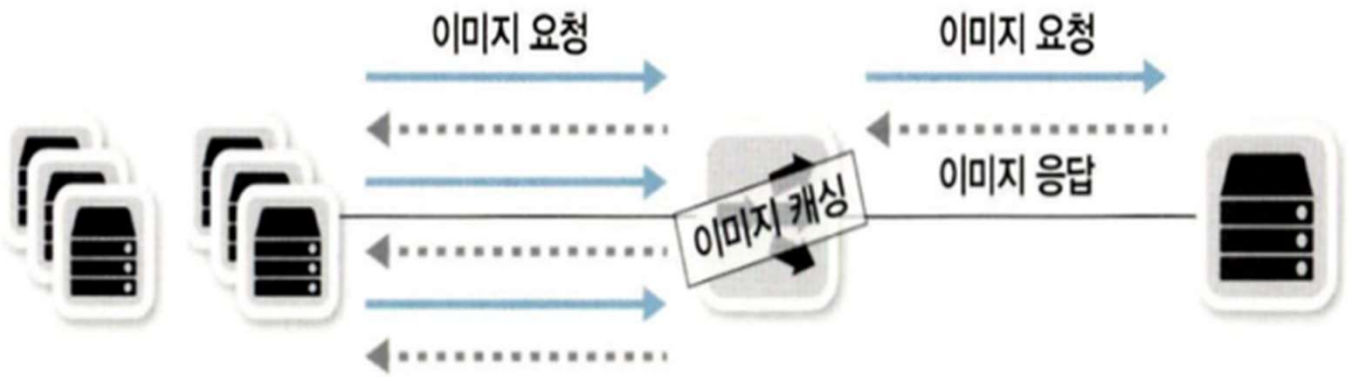
4계층 장비

- ❖ Load Balancer
  - ✓ L4 Switch & ADC
    - ● L4 Switch
      - ◆ L4 스위치는 4계층에서 동작하면서 TCP, UDP 정보를 기반으로 부하를 분산함
      - ◆ 부하 분산 뿐 만 아니라 TCP 계층에서의 최적화와 보안 기능도 함께 제공할 수 있음
      - ◆ TCP 레벨의 간단한 Denial of Service(서비스 거부 공격) 공격을 방어하거나 서버 부하를 줄이기 위해 TCP 세션 재사용과 같이 보안과 성능을 높여주는 기능도 함께 제공할 수 있음
      - ◆ L4 스위치의 서버 성능 향상 기법



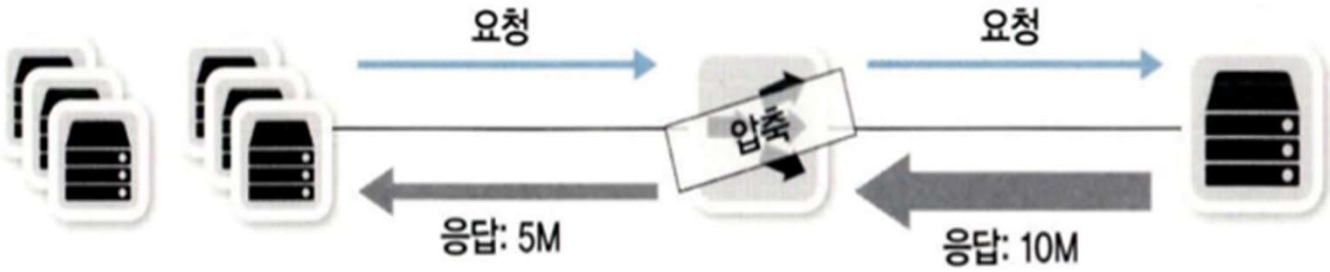
4계층 장비

- ❖ Load Balancer
  - ✓ L4 Switch & ADC
    - ● ADC
      - ◆ ADC는 애플리케이션 프로토콜을 이해하고 애플리케이션 내용에 대한 분산, Redirection, 최적화를 제공해 L4 스위치보다 더 다양한 기능을 사용할 수 있음
      - ◆ ADC는 성능 최적화를 위해 서버에서 수행하는 작업 중 부하가 많이 걸리는 작업을 별도로 수행하는데 그 중 하나가 이미지나 정적 콘텐츠 Caching 기능



4계층 장비

- ❖ Load Balancer
  - ✓ L4 Switch & ADC
    - ● 웹 서버에는 콘텐츠 압축 기능이 있지만 ADC에서 이 역할을 수행해 웹 서버의 부하를 줄일 수 있는데 ADC는 하드웨어 가속이나 소프트웨어 최적화를 통해 이런 부하가 걸리는 작업을 최적화하는 기능이 있음



4계층 장비

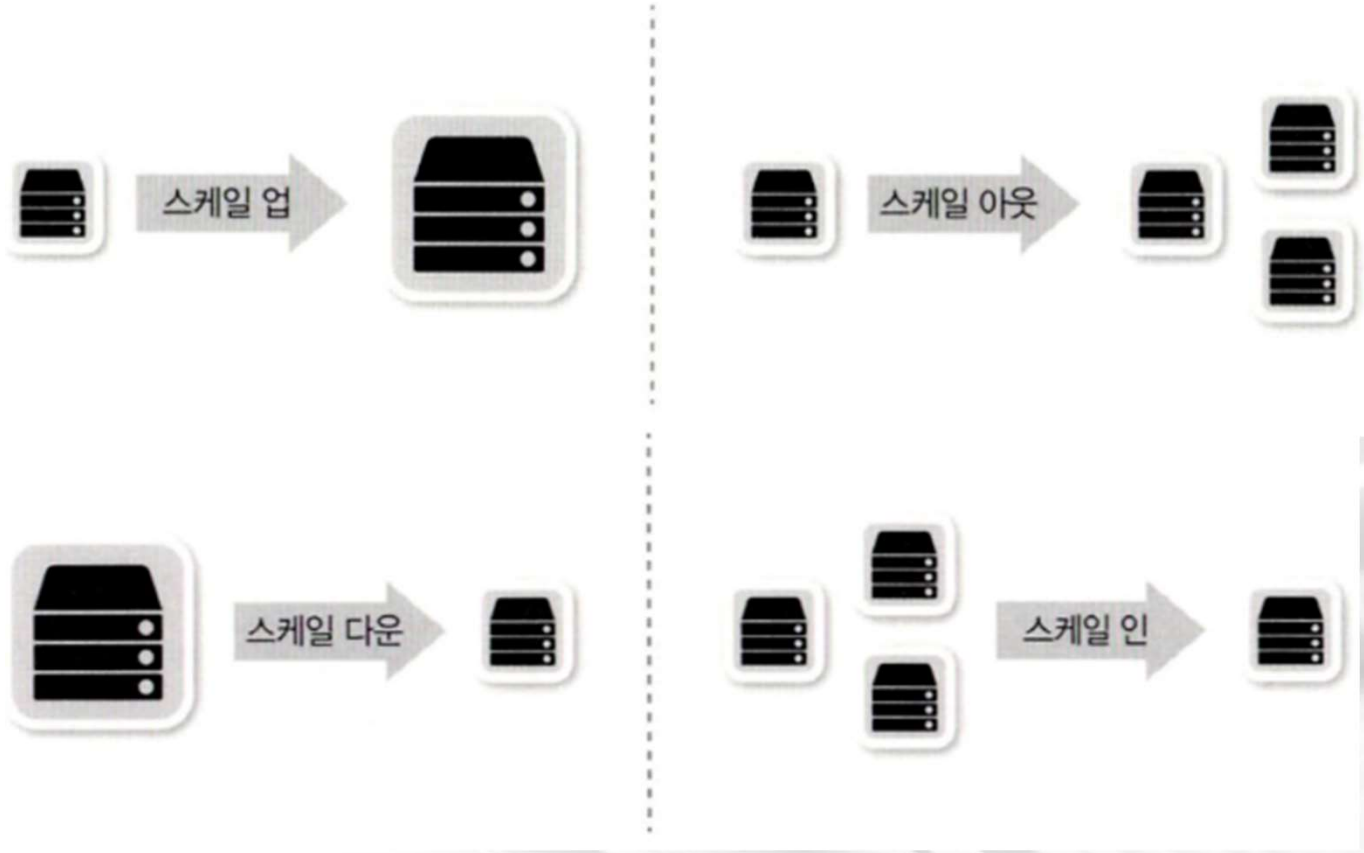
- ❖ Load Balancer
  - ✓ L4 Switch & ADC
    - 최근 SSL 프로토콜을 사용하는 비중이 늘면서 웹 서버에 SSL 암호화 및 복호화 부하가 늘고 있음
    - 개인정보 보호를 위해 개인정보가 전달되는 일부 페이지에서만 SSL을 사용해왔지만 보안 강화를 위해 웹사이트 전체를 SSL로 처리하는 추세
    - ADC에서는 SSL의 End Point로 동작해 클라이언트에서 ADC까지 구간을 SSL로 처리해 주고 ADC와 웹 서버 사이를 일반 HTTP를 이용해 통신하는데 이런 기능을 사용할 때는 웹 서버 여러 대의 SSL 통신을 하나의 ADC에서 수용하기 위해 ADC에 전용 SSL 가속 카드를 내장

4계층 장비

❖ Load Balancer

✓ 시스템 확장 방법

- 서비스를 운영하다 보면 시스템 하나로 모든 서비스를 감당할 수 없을 만큼 성장하는 경우가 생기는데 데이터 양이 늘거나 CPU나 메모리 사용량이 늘어 서버 하나로 서비스가 불가능해지면 시스템을 확장

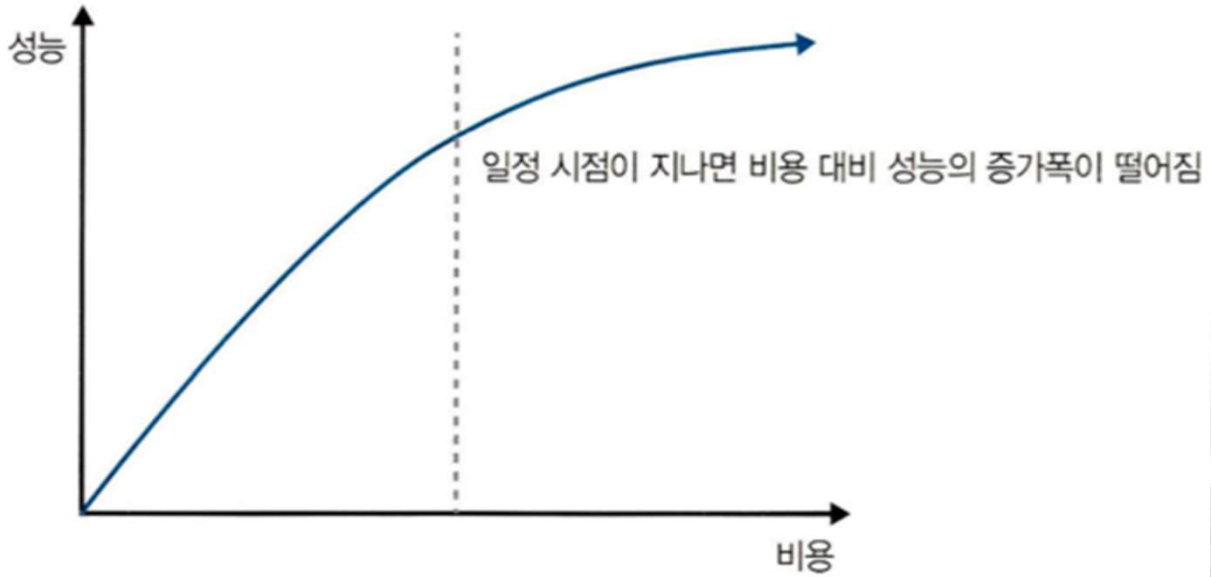


4계층 장비

- ❖ Load Balancer
  - ✓ 시스템 확장 방법
    - ● 스케일 업
      - ◆ 시스템을 가장 쉽게 확장해 서비스 용량을 키우는 방법은 스케일 업인데 기존 시스템에 CPU, 메모리, 디스크와 같은 내부 컴포넌트 용량을 키우거나 이것이 불가능할 경우 새로 더 큰 용량의 시스템을 구매해 서비스를 옮기는 방법인데 스케일 업이 가능한 요소나 서비스가 있고 그렇지 않은 경우가 있음
      - ◆ 파일 저장소인 디스크는 어느 정도까지는 쉽게 확장할 수 있지만 CPU, 메모리를 확장하기는 쉽지 않음
      - ◆ 스케일 업을 고려해 CPU 여러 개를 꽂을 수 있고 메모리 뱅크를 많이 가진 보드를 구매해야 하는데 이 경우 초기 투자비용이 커짐
      - ◆ 스케일 업은 확장을 미리 고려해 시스템을 구축하면 초기 비용이 커지고 서비스에 적합한 시스템을 구매하면 확장이 필요할 때 기존 시스템을 버리고 더 큰 시스템을 새로 구매하므로 기존 시스템 비용이 낭비되는 문제가 있음
      - ◆ 스케일 업은 필요한 용량만큼 시스템을 늘리는 데 비용이 기하급수적으로 증가하는 문제가 있는데 저가형 CPU보다 성능이 2배 우수한 CPU는 10배 이상의 비용이 필요할 수 있음

4계층 장비

- ❖ Load Balancer
  - ✓ 시스템 확장 방법
    - ● 스케일 업
      - ◆ 스케일 업 과 성능 곡선



4계층 장비

- ❖ Load Balancer
  - ✓ 시스템 확장 방법
    - ● 스케일 아웃
      - ◆ 스케일 업은 시스템 하나의 용량 자체를 키우지만 스케일 아웃은 같은 용량의 시스템을 여러 대 배치
      - ◆ 사용자 천 명의 요청을 처리하는 시스템이 한 대라면 5천 명을 위해서는 5대의 시스템을 병렬로 운영하는 방법
      - ◆ 스케일 아웃을 위해 새로 시스템 설계를 하거나 분산을 위한 별도의 프로세스를 운영하거나 Load Balancer와 같이 부하를 분산해주는 별도의 외부 시스템이 필요

4계층 장비

❖ Load Balancer

✓ 시스템 확장 방법

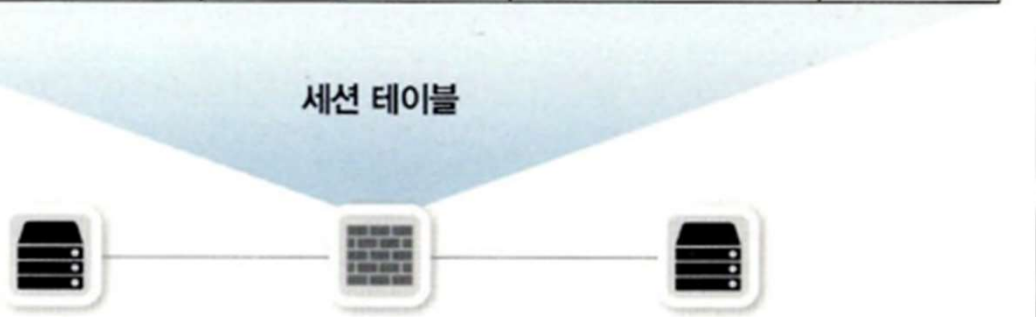
● 장단점 비교

	스케일 업(Scale-Up)	스케일 아웃(Scale-Out)
설명	하드웨어 성능 자체를 업그레이드하거나 더 높은 성능의 시스템으로 마이그레이션하는 방법	여러 대의 서버로 로드를 분산하는 방법. 서비스 자체를 구분해 나누거나 같은 서비스를 분산해 처리하는 방법이 있다.
장점	부품을 쉽게 추가할 수 있으면 시스템 설계 변경 없이 서비스 사용량을 쉽게 늘릴 수 있다(주로 기존 대형 유닉스 시스템에서 사용함)	스케일 업 방식보다 더 적은 비용으로 시스템 확장이 가능하다. 여러 대의 시스템에 로드를 적절히 분산해 하나의 시스템에 장애가 발생하더라도 서비스에 미치는 영향이 없도록 결함허용(Fault Tolerance)을 구현할 수 있다.
단점	부품 추가가 어렵다(최근 x86). 시스템이 커질수록 비용이 기하급수적으로 증가한다.	스케일 아웃을 위해 별도의 복잡한 아키텍처를 이해하고 운영해야만 할 수 있다. 프로세스나 네트워크 장비가 추가로 필요할 수 있다.

4계층 장비

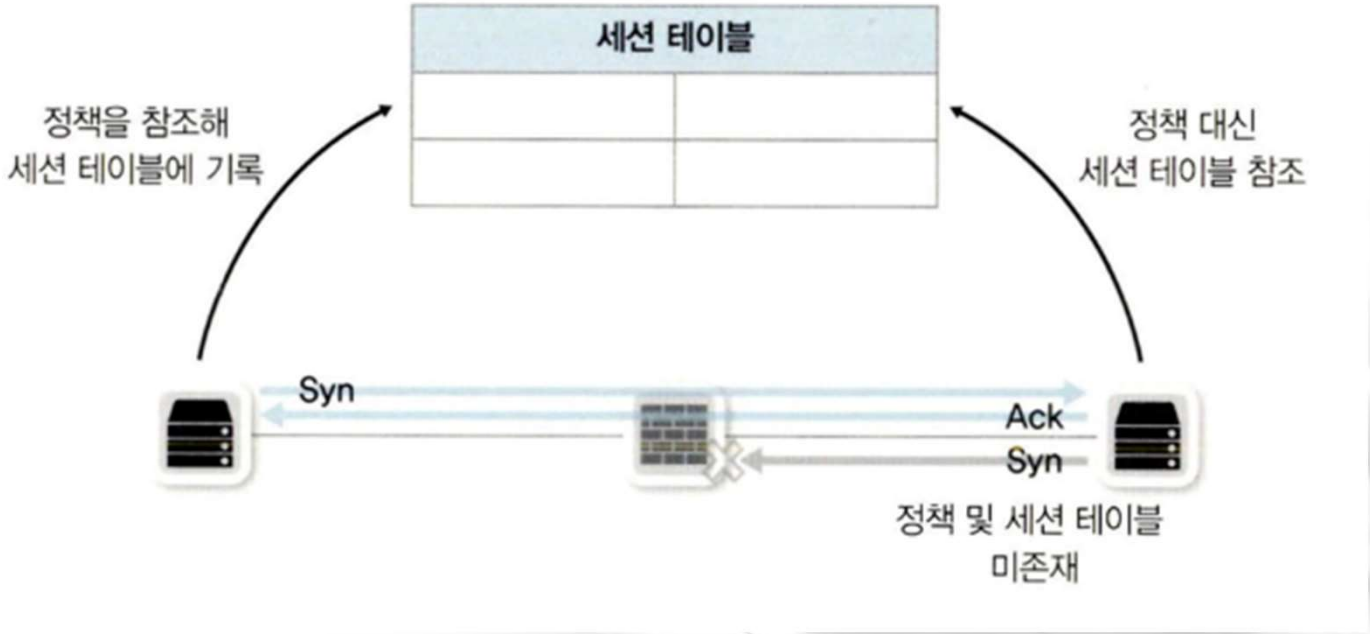
- Firewall
  - ✓ 네트워크 중간에 위치해 해당 장비를 통과하는 트래픽을 사전에 주어진 정책 조건에 맞추어 허용(Permit)하거나 차단(Deny)하는 장비
  - ✓ 네트워크에서 보안을 제공하는 장비를 넓은 의미에서 모두 방화벽의 일종으로 불러왔지만 일반적으로 네트워크 3, 4 계층에서 동작하고 세션을 인지, 관리하는 SPI(Stateful Packet Inspection) 엔진을 기반으로 동작하는 장비를 방화벽이라고 부름
  - ✓ 세션 테이블이 있는 방화벽

출발지 주소	목적지 주소	출발지 서비스 포트	목적지 서비스 포트	...
1.1.1.10	10.10.10.11	30513	80	



4계층 장비

- Firewall
  - ✓ 방화벽은 NAT(Network Address Translation) 동작 방식과 유사하게 세션 정보를 장비 내부에 저장하고 패킷이 외부로 나갈 때 세션 정보를 저장하고 패킷이 들어오거나 나갈 때 저장했던 세션 정보를 먼저 참조해 들어오는 패킷이 외부에서 처음 시작된 것인지 내부 사용자가 외부로 요청한 응답인지 구별



4계층 장비

❖ 세션 관리 – 세션 테이블 유지 및 세션 정보 동기화

✓ 개요

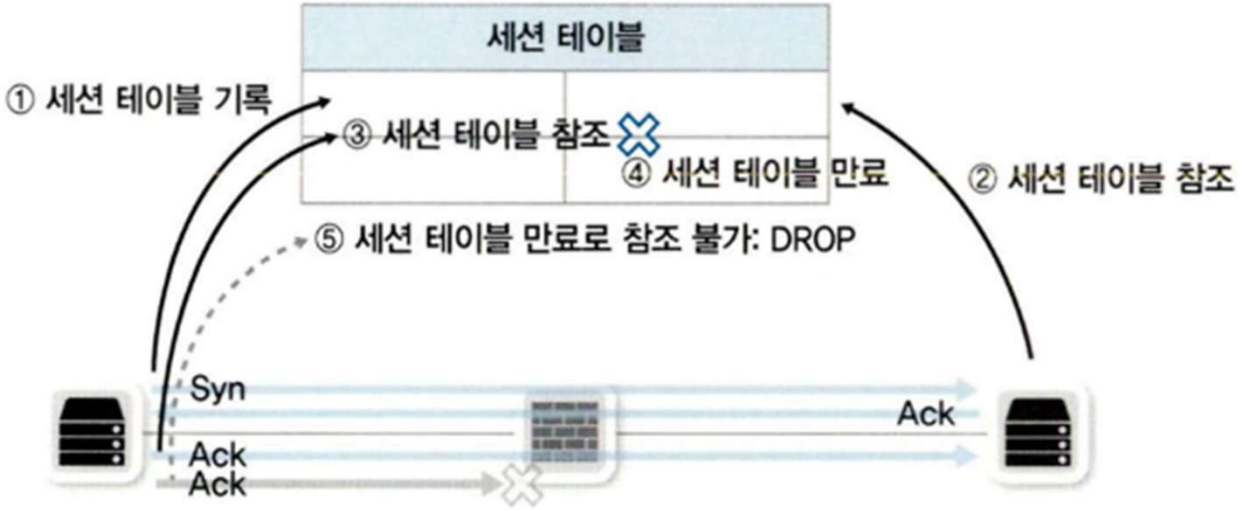
- 종단 장비에서 통신을 시작하면 중간에 있는 세션 장비는 세션 상태를 테이블에 기록
- 통신이 없더라도 종단 장비 간 통신이 정상적으로 종료되지 않았다면 일정 시간 동안 세션 테이블을 유지해야 하는데 하지만 이런 세션 테이블은 메모리에 저장되므로 메모리 사용률을 적절히 유지하기 위해 일정 시간만 세션 정보를 저장
- 악의적인 공격자가 과도한 세션을 발생시켜 정상적인 세션 테이블 생성을 방해하는 세션 공격으로부터 시스템을 보호하기 위해 타임아웃값을 더 줄이기도 함
- 여러 가지 이유로 세션 장비는 세션 정보를 무제한으로 저장할 수 없고 여러가지 애플리케이션 통신을 관리하므로 일반적인 애플리케이션에 맞추어 적당한 세션 타임아웃값을 유지하지만 일부 애플리케이션은 세션을 한 번 연결해놓고 다음 통신이 시도될 때까지 세션이 끊기지 않도록 세션 타임아웃값을 길게 설정하기도 하는데 이런 종류의 애플리케이션이 통신할 때 장비의 세션 타임아웃값이 애플리케이션의 세션 타임아웃값보다 짧으면 통신에 문제가 생김
- 중간 세션 장비의 세션 유지 시간이 지나 세션 테이블에 있는 세션 정보가 사라졌는데도 양쪽 단말에서는 세션이 유지되고 있다면 다시 통신이 시작되어 데이터를 보낼 때 중간 세션 장비에서 막히는 문제가 발생
- 세션 장비의 세션 테이블에 세션이 없는 상황에서 SYN이 아닌 ACK로 표시된 패킷이 들어오면 세션 장비에서는 비정상 통신으로 판단해 패킷을 차단하고 그런 종류의 패킷을 통과시키는 옵션을 설정해 패킷을 강제로 통과시키더라도 반대 방향으로 데이터가 들어오면 정책에 막힐 수 있음

4계층 장비

❖ 세션 관리 – 세션 테이블 유지 및 세션 정보 동기화

✓ 개요

- 세션 장비의 세션 만료 시간이 애플리케이션 세션 만료 시간보다 짧을 경우(세션 설정 후 일정시간 통신이 없으면 세션이 만료)



4계층 장비

- ❖ 세션 관리 – 세션 테이블 유지 및 세션 정보 동기화
  - ✓ 문제 해결책 – 세션 장비 운영자 입장
    - ● 세션 만료 시간 증가
      - ◆ 세션 장비 운영자가 애플리케이션에 맞게 세션 만료 시간을 늘리는 방법
      - ◆ 애플리케이션의 세션 유지 시간보다 방화벽 세션 유지 시간이 길게 설정
      - ◆ 이 경우 대부분의 세션 장비는 포트 번호나 IP 주소 마다 별도의 세션 만료 시간을 설정할 수 있어 전체 세션 유지 시간이 길어져 시스템 메모리가 고갈되는 문제를 예방할 수 있음
      - ◆ 세션 장비 운영자가 정확한 정보를 얻어 설정할 수 있도록 애플리케이션측 개발자나 관리자가 애플리케이션 고유의 세션 유지 시간을 미리 알려주어야 함
    - ● 세션 시간을 둔 채로 중간 패킷을 수용할 수 있도록 방화벽 설정(세션 장비 중 방화벽에 해당)
      - ◆ 세션 테이블에 정보가 없는 ACK 패킷이 방화벽에 들어오면 방화벽은 패킷을 차단하지만 방화벽 옵션을 조정해 세션 테이블에 정보가 없는 ACK 패킷이 들어오더라도 세션을 새로 만들어 통과시킬 수 있는 옵션을 활용
      - ◆ 이런 해결책은 전체적인 보안이 취약해지는 기능이므로 여러가지 고려가 필요하고 가능하면 적용하지 않는 것을 권장

4계층 장비

- ❖ 세션 관리 – 세션 테이블 유지 및 세션 정보 동기화
  - ✓ 문제 해결책 – 세션 장비 운영자 입장
    - ● 세션 장비에서 세션 타임아웃 시 양 단말에 세션 종료 통보
      - ◆ 양 종단 장비의 세션 정보와 중간 세션 장비의 세션 정보가 일치하지 않아 발생하는 문제를 해결하기 위해 사용하는 기능
      - ◆ 세션 상태 정보를 강제로 공유하기 위해 세션 장비에서는 세션 타임아웃 시 세션 정보를 삭제하는 것이 아니라 세션 정보에 있는 양 종단 장비에 세션 정보 만료(RST)를 통보
      - ◆ TCP의 RST 플래그를 1로 세팅해 양 종단 장비에 전송하면(A를 출발지로 B를 도착지로 B를 출발지로 A를 도착지로) 양 종단 장비에서는 세션이 비정상적으로 종료된 것으로 판단해 해당 세션을 끊고 애플리케이션에서 통신이 필요하면 새로운 세션을 맺어 통신

4계층 장비

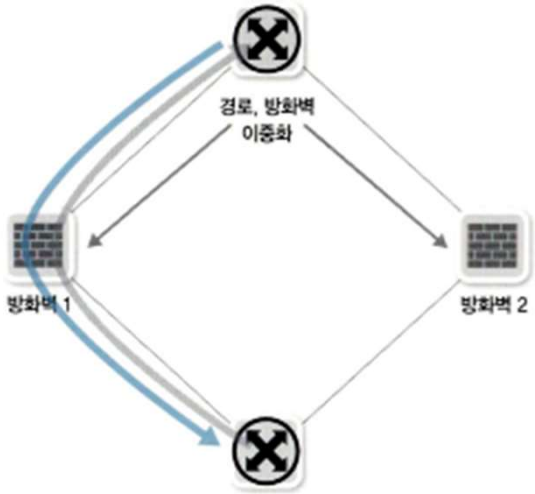
- ❖ 세션 관리 – 세션 테이블 유지 및 세션 정보 동기화
  - ✓ 문제 해결책 – 개발자 입장
    - ● 애플리케이션에 주기적인 패킷 발생 기능 추가
      - ◆ 애플리케이션과 세션 장비의 세션 타임아웃 시간을 일치시키는 가장 좋은 방법은 애플리케이션에서 패킷을 주기적으로 발생시키는 것
      - ◆ 애플리케이션 개발 시 중간에 통신이 없더라도 일정시간마다 양 단말 애플리케이션의 세션 상태 정보를 체크하는 더미 패킷(Dummy Packet)을 보내는 기능을 추가하면 패킷이 주기적으로 발생해 중간 방화벽에서 세션 타임아웃이 발생하기 전에 세션을 유지할 수 있음
      - ◆ 최근 대부분의 플랫폼에서는 이런 기능들을 내장하고 개발하도록 안내하고 있음
      - ◆ 중간 세션 장비의 세션 만료 시간으로 인한 문제를 해결하는 가장 바람직한 방법

4계층 장비

❖ 비대칭 경로 문제

✓ 개요

- 네트워크의 안정성을 높이기 위해 네트워크 회선과 장비를 이중화
- 이중화 한 경우 패킷이 지나가는 경로가 2개 이상이므로 들어오는 패킷과 나가는 패킷의 경로가 같거나 다를 수 있음
- 들어오는 패킷과 나가는 패킷이 같은 장비를 통과하는 것을 대칭 경로(Symmetric Path)라 하고 다른 장비를 통과하는 것을 비대칭 경로(Asymmetric Path)
- 대칭 경로: 통신에 문제가 없음

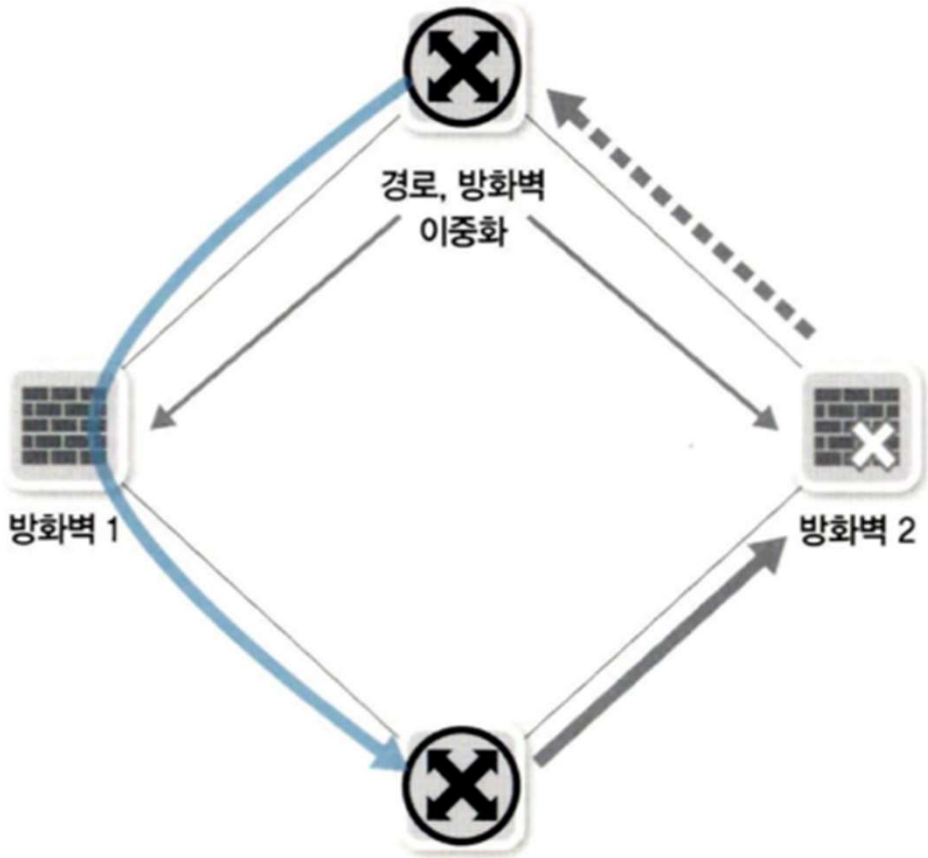


4계층 장비

❖ 비대칭 경로 문제

✓ 개요

- 비대칭 경로: 들어오는 패킷과 나가는 패킷이 한 장비를 통과하지 않아서 패킷이 드롭될 수 있음



4계층 장비

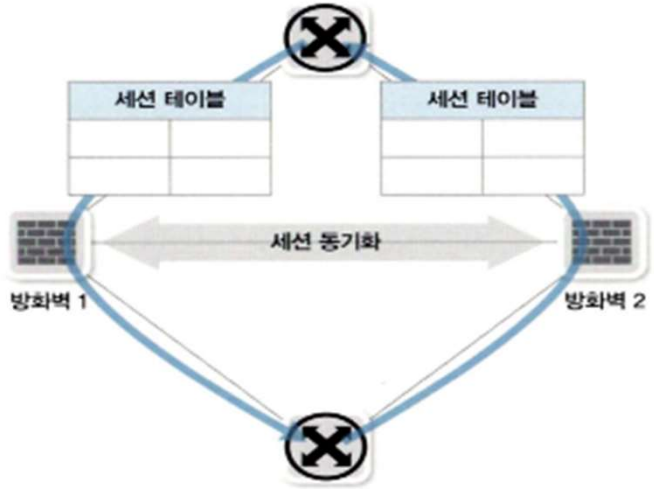
❖ 비대칭 경로 문제

✓ 개요

- 2, 3 계층 네트워크 장비는 세션을 고려할 필요가 없기 때문에 네트워크 엔지니어 대다수가 세션 장비의 이런 특징을 파악하지 못하고 네트워크 효율성에만 초점을 맞추어 비대칭 경로를 사용하도록 네트워크를 디자인하는 경우가 많음
- 비대칭 경로 문제의 가장 좋은 해결 방법은 비대칭 경로가 생기지 않도록 네트워크와 경로를 디자인하는 것
- 어쩔 수 없이 비대칭 경로가 생기면 세션 장비에 이런 비대칭 경로를 처리하는 기능을 이용할 수 있지만 이런 비대칭 패킷을 처리하기 위한 노력이 필요하여 세션 장비의 성능이 저하되거나 보안이 약화될 수 있음

4계층 장비

- ❖ 비대칭 경로 문제
  - ✓ 해결책
    - ● 세션 테이블 동기화
      - ◆ 세션 테이블을 동기화하면 두 개 경로상의 두 장비가 하나의 장비처럼 동작하므로 비대칭 경로에서도 정상적으로 동작할 수 있음
      - ◆ 패킷 경로를 변경하지 않고 동작한다는 장점이 있지만 세션을 동기화하는 시간보다 패킷 응답이 빠르면 정상적으로 동작하지 않을 수 있다는 단점이 있음
      - ◆ 데이터 센터에서 응답시간이 빠른 애플리케이션을 사용할 경우 세션 동기화 시간보다 응답시간이 더 빠를 수 있으므로 이 기능을 사용하는 것을 추천하지 않음
      - ◆ 이 기능은 응답시간이 비교적 긴 인터넷 게이트웨이로 방화벽이 사용될 때 유용하게 사용될 수 있음



4계층 장비

- ❖ 비대칭 경로 문제
  - ✓ 해결책
    - ● 세션 장비에서 보정
      - ◆ 나가는 패킷이 통과하지 않았는데 패킷이 장비로 들어온 경우 나가는 패킷이 통과한 다른 세션 장비 쪽으로 패킷을 보내 경로를 보정
      - ◆ 강제로 대칭 경로를 만들어 주므로 비대칭 경로로 인한 문제를 해결할 수 있음
      - ◆ 강제로 다른 방화벽으로 패킷을 보내기 위해 방화벽 간 통신용 링크가 필요하고 MAC 주소를 변경하는 MAC Rewriting 이나 기존 패킷에 MAC 주소를 한 번 더 캡슐화하는 Tunneling기법으로 경로를 보정
      - ◆ 경로 보정