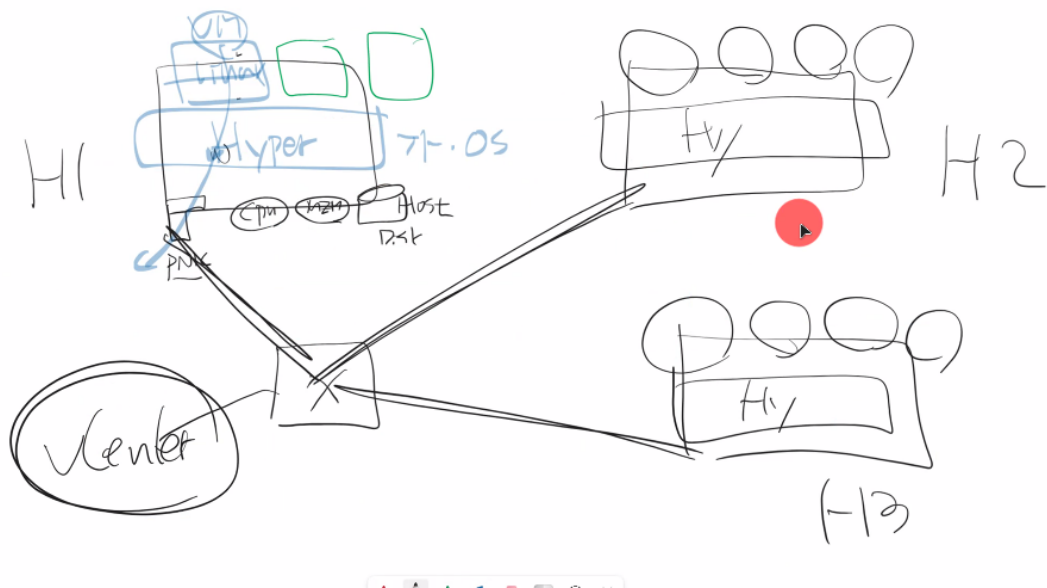


VM웨어 5일차

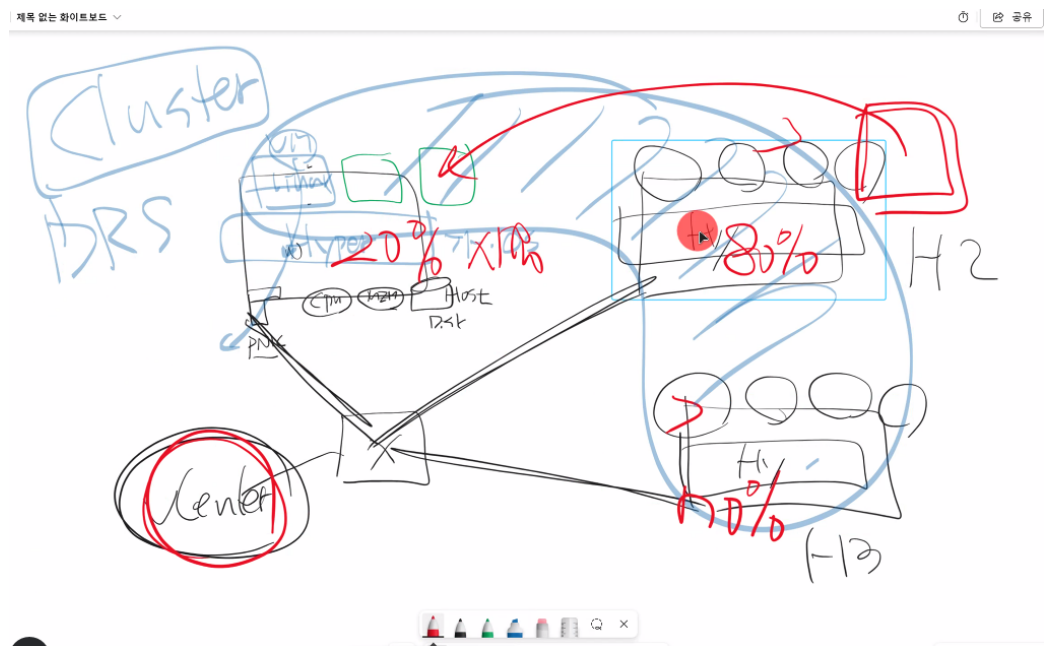
- 주제정하기(시스코 IOS)
 - 주제설명
 - 사용목적
 - 시연(구현)
 - 결과확인
- IPv6 에서의 ARP
- VLAN 등등..

vCenter



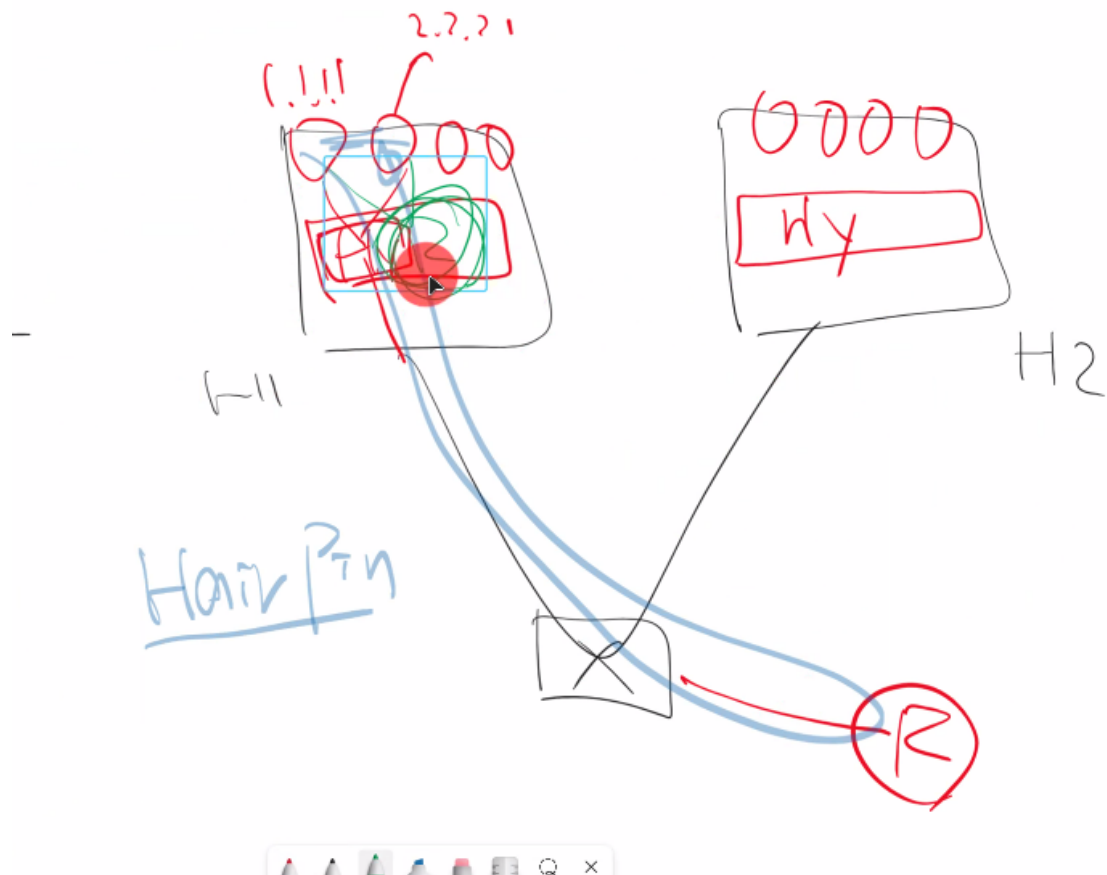
- 하이퍼바이저, 호스트, 가상머신 관리
- 각 호스트를 유지보수 해야한다.

- 근데 서비스에 영향을 받는다.
- 실시간으로 다른 호스트에게 마이그레이션한다.
- 그리고 스케일업하고 다시 원래 호스트로 마이그레이션
 - vm웨어가 이 기능 때문에 먹고산다
- 클러스터로 묶으면 할 수 있는기능
 - DRS(분산 리소스 스케줄러)



- 자원을 덜 쓰는 곳에 VM을 자동으로 넘기는 기능 DRS
- HA High avility 고가용성도 있다.

네트워크 가상화



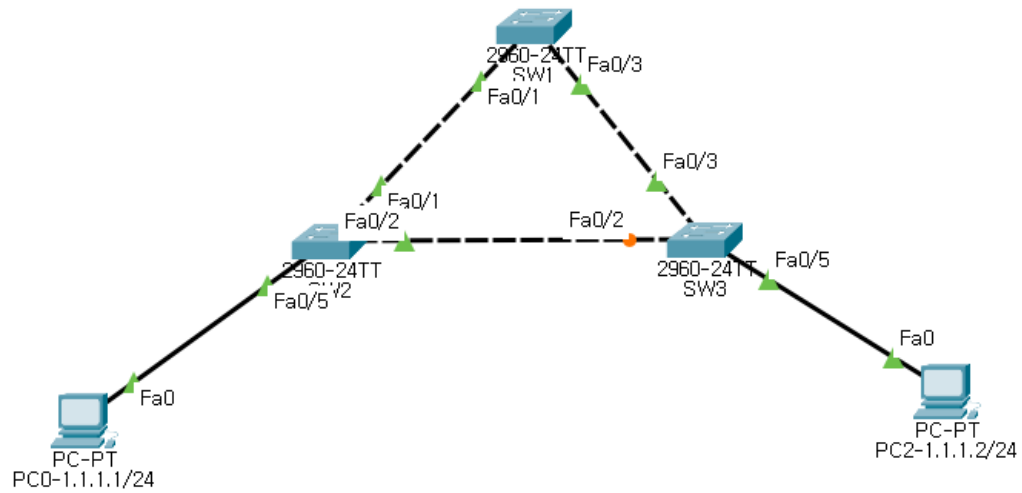
- 1.1.1.1과 2.2.2.1 통신하려는데 라우터 필요. 근데 트래픽이 헤어핀처럼 돌고 돌아서 트래픽이 들어감
- 이는 경로가 효율화되어 있지 않음 이를 해결하기 위해서 네트워크 가상화가 필요하다.
 - NSX (하이퍼바이저 필요)

STP 프로토콜

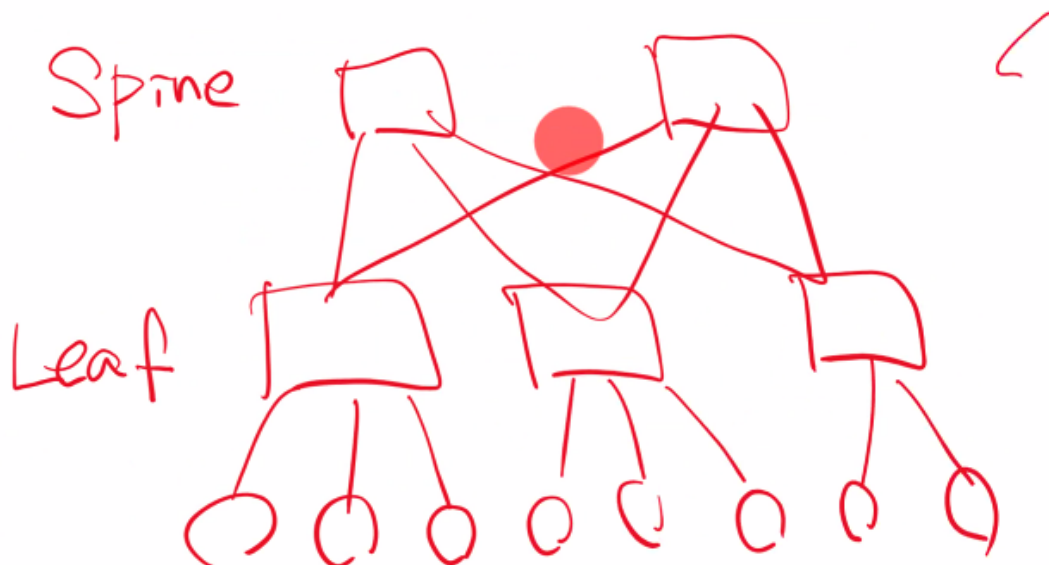
- GPT에 있는내용 정리 L2 레벨에서 플러딩 제어
- 무한 루프 제어

spanning tree

- bck 상태이었던것도 하나가 섰다운되면 다시 Designated Port 가된다.



- 폐구조를 임의로 만들고 block port를 찾는다.
- 지금 블록포트는 0/2인데 0/2에 트래픽이 많다면? 루트 포트를 0/2로 바꿔주고 0/3을 block 시키는게 네트워크 엔지니어의 일이다.

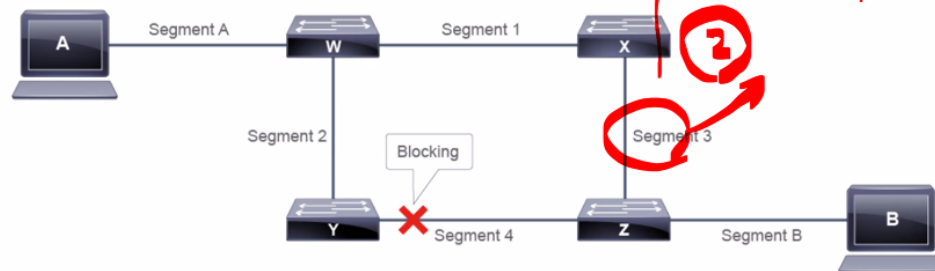


- 리프 스위치, 스파인 스위치?

Spanning Tree Operation

Spanning Tree Operation has the following characteristics:

- It provides a loop-free redundant network topology by placing certain ports into a blocking state.
- It is published in the IEEE 802.1D specification.



- 세그먼트 용어정리
 - L4 계층의 데이터구조 TCP
 - 스페닝 트리의 네트워크 대역을 세그먼트라고함.

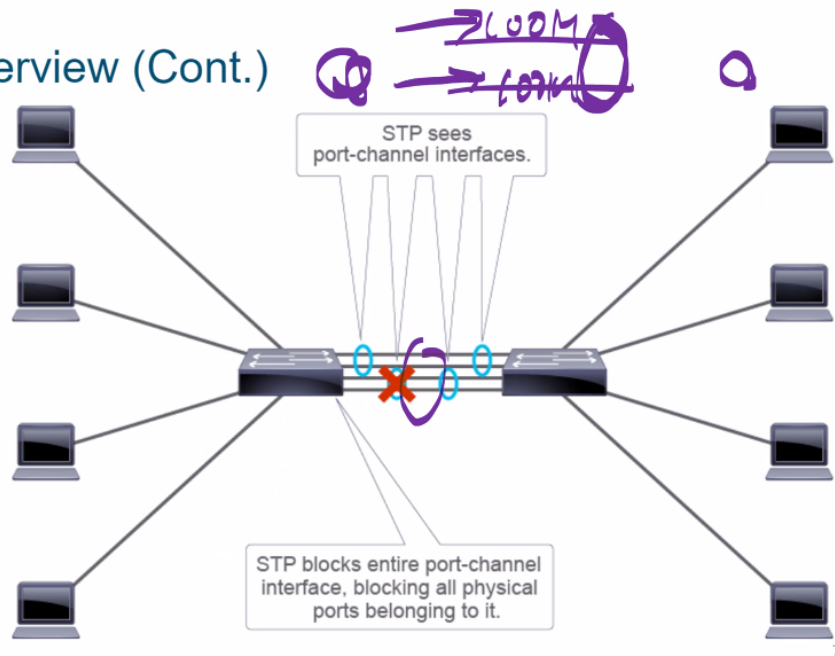
EtherChannel Overview (Cont.)

EtherChannel has the following characteristics:

- Logical aggregation of links between switches
- High bandwidth
- Load sharing across links
- One logical port to STP
- Redundancy



L B



- 이더채널은 여러 링크를 하나로 묶어서 사용한다.
- 이 묶음은 STP프로토콜이 하나의 링크로 인식한다.
- 로드 셰어링 기능과 고가용성이 보장된다. 4개중 1개가 죽어도 3개로 계속 가게되어 STP를 재구성할 필요가 없다.
- 그래서 고성능, 고안정성을 제공한다.
- 밴드넓이까지 넓어짐

EtherChannel 구성 방법

- **LACP (Link Aggregation Control Protocol):** LACP는 IEEE 802.3ad 표준에 기반한 프로토콜로, 여러 스위치 벤더 간의 호환성을 제공합니다. LACP는 링크 상태를 모니터링하고, 동적으로 링크를 추가하거나 제거할 수 있도록 합니다.
- **PAGP (Port Aggregation Protocol):** Cisco 고유의 EtherChannel 프로토콜로, LACP와 유사하게 링크 상태를 모니터링하며 동적으로 링크를 관리할 수 있습니다.
- **수동 모드:** EtherChannel은 고정된 설정으로, 수동으로 구성된 물리적 링크를 묶을 수 있습니다. 이 모드는 LACP나 PAGP를 사용하지 않고, 모든 링크를 고정된 상태로 유지합니다.

EtherChannel 구성하는 CLI 명령어



Cisco 패킷 트레이서에서 EtherChannel을 구성하기 위해 사용할 수 있는 CLI 명령어의 예시를 단계별로 설명하겠습니다. 이 예시는 두 개의 스위치(Switch1, Switch2) 사이에서 EtherChannel을 구성하는 상황을 가정하고, EtherChannel을 통해 여러 물리적 인터페이스를 논리적으로 묶는 방법을 보여줍니다.

여기에서는 EtherChannel을 구성할 때 많이 사용되는 ****LACP(Link Aggregation Control Protocol)****를 사용하여 EtherChannel을 설정하는 방법을 설명하겠습니다.

단계 1: 스위치 인터페이스를 EtherChannel 그룹에 할당하기

1.1 Switch1에서 EtherChannel 설정

1. 스위치 설정 모드로 이동:

```
enable
configure terminal
```

2. 인터페이스 범위를 지정하여 EtherChannel을 설정할 물리적 인터페이스를 선택합니다. 예를 들어, GigabitEthernet 0/1과 0/2를 사용할 경우:

```
interface range GigabitEthernet 0/1 - 2
```

3. LACP 모드로 EtherChannel 그룹 생성 (여기서는 그룹 번호 1을 사용):

```
channel-group 1 mode active
```

- **mode active**: LACP를 사용하며, 활성 모드로 설정합니다. 활성 모드는 해당 포트가 LACP를 시작할 수 있도록 합니다.

4. EtherChannel에 설명을 추가하고, 설정이 끝났음을 알리기 위해 종료합니다:

```
description Link to Switch2 using EtherChannel
exit
```

1.2 Switch2에서 EtherChannel 설정

Switch2에서도 동일한 과정을 반복하여 EtherChannel을 설정합니다:

1. 스위치 설정 모드로 이동:

```
enable  
configure terminal
```

2. 인터페이스 범위를 지정하여 EtherChannel을 설정할 물리적 인터페이스를 선택합니다:

```
interface range GigabitEthernet 0/1 - 2
```

3. LACP 모드로 EtherChannel 그룹 생성 (Switch1에서 사용한 그룹 번호와 동일한 번호 사용):

```
channel-group 1 mode active
```

4. EtherChannel에 설명을 추가하고, 설정이 끝났음을 알리기 위해 종료합니다:

```
description Link to Switch1 using EtherChannel  
exit
```

단계 2: EtherChannel 인터페이스를 설정하기

1. Switch1과 Switch2에서 각각의 스위치에서 논리적 인터페이스로 이동합니다. 이 인터페이스는 **Port-channel** 로 지정되며, 그룹 번호를 따릅니다. 여기서는 **Port-channel 1** 을 사용합니다:

Switch1:

```
interface Port-channel 1
```

Switch2:

```
interface Port-channel 1
```

2. 이 인터페이스에 대한 설정을 진행할 수 있습니다. 예를 들어, 트렁크 모드 설정:

- Switch1과 Switch2에서 각각:


```
switchport mode trunk
```

3. VLAN을 특정할 수도 있습니다. 예를 들어, 네이티브 VLAN 설정:

```
switchport trunk native vlan 10
```

4. 설정 종료:

```
exit
```

단계 3: 설정 검증

EtherChannel 구성이 완료된 후, 각 스위치에서 구성을 검증합니다.

1. EtherChannel 요약 정보 확인:

```
show etherchannel summary
```

이 명령어는 EtherChannel의 상태와 구성된 포트에 대한 요약 정보를 표시합니다. 정상적으로 설정되었다면, `Port-channel 1` 인터페이스가 `up` 상태로 표시되고, 물리적 인터페이스들도 `P` (포트가 EtherChannel의 일부임을 나타냄) 상태로 표시됩니다.

2. 상세 상태 확인:

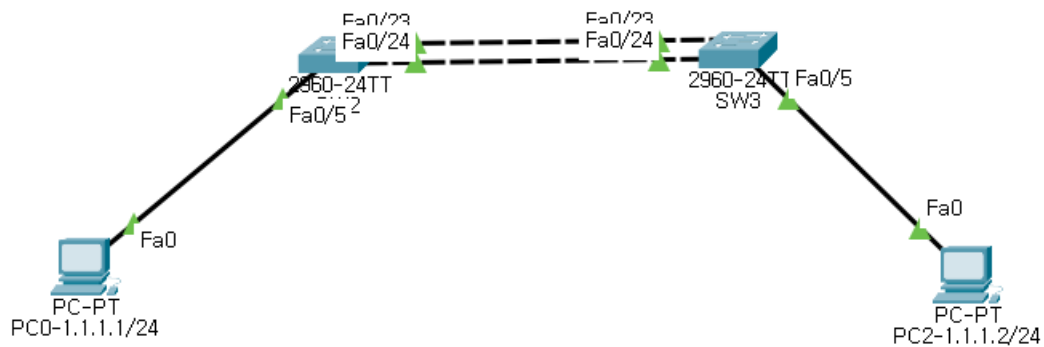
```
show etherchannel port-channel
```

이 명령어는 EtherChannel 포트 채널의 상세 정보를 확인할 수 있으며, 각 링크의 상태, 사용 중인 로드 밸런싱 방법, LACP 상태 등을 보여줍니다.

요약

- EtherChannel을 구성하려면 물리적 인터페이스를 `channel-group` 명령어를 통해 그룹으로 묶습니다.
- `mode active` 옵션을 사용하여 LACP를 활성화합니다.
- `Port-channel` 인터페이스에서 트렁크 모드 등 추가 설정을 할 수 있습니다.
- 설정이 완료된 후 `show etherchannel summary` 와 같은 명령어를 사용하여 설정이 제대로 되었는지 확인합니다.

이 CLI 명령어 예시를 통해 Cisco 패킷 트레이서에서 EtherChannel을 성공적으로 구성할 수 있습니다. 추가적인 질문이 있으시면 언제든지 말씀해 주세요!



```
*** LSW, RSW ***
en
conf t
int ran f0/23-24
  channel-group 3 mode acti
```

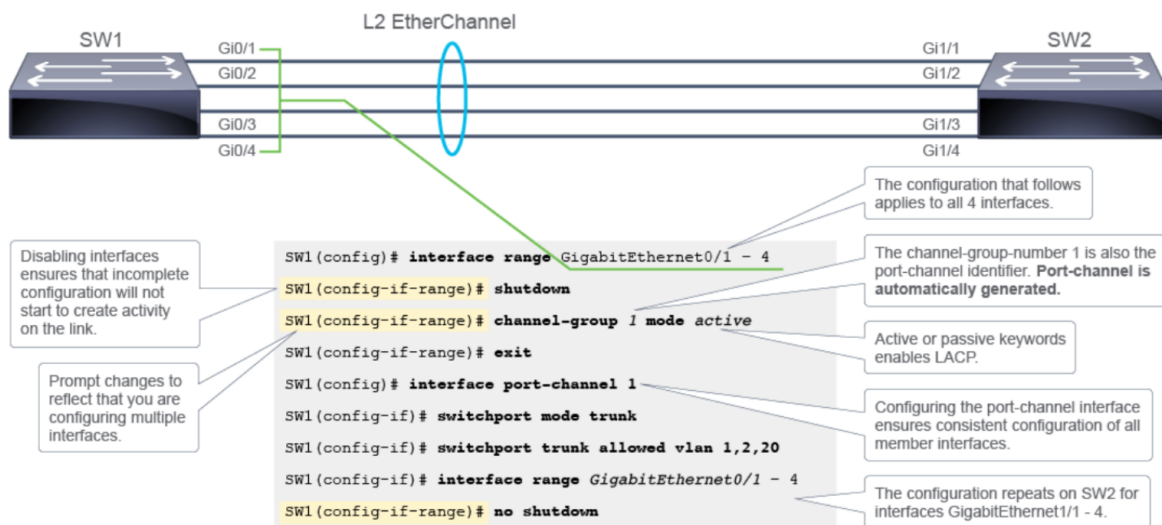
```
*** LSW, RSW ***
en
conf t
int ran f0/23-24
  channel-group 3 mode acti
!
int port-channel 3
  switchport mode trunk
```

```
// etherchannel 정보확인
do sh etherchannel summary
```

```
// spanning 정보확인
do sh span
```

L2 스위치 이더채널 구성

Configuring EtherChannel



L3 스위치 이더채널 구성

Configuring EtherChannel (Cont.)

A logical port channel interface identified by the number 3 is created.

The configuration repeats on SW2 for interfaces GigabitEthernet1/1 - 4.

```

SW1(config)# interface port-channel 3
SW1(config-if)# no switchport
SW1(config-if)# ip address 172.16.3.10 255.255.255.0
SW1(config-if)# interface range GigabitEthernet0/1 - 4
SW1(config-if-range)# no switchport
SW1(config-if-range)# channel-group 3 mode on
SW1(config-if-range)# exit
    
```

no switchport command turns interface into a routed interface.

IP address is assigned to the routed port-channel 3 interface.

The configuration that follows applies to all 4 interfaces.

no switchport command turns all member interfaces into a routed interface.

on keyword manually established the aggregated link and is used on platforms that do not support LACP.

13

Vlan 이더채널 구성하기(미션)

1. 내부통신 가능하게... (router on a stick 방식 사용)
- sub-if 번호는 vlan 번호로

2. 내부 <-> 외부 통신 가능하게... (static)

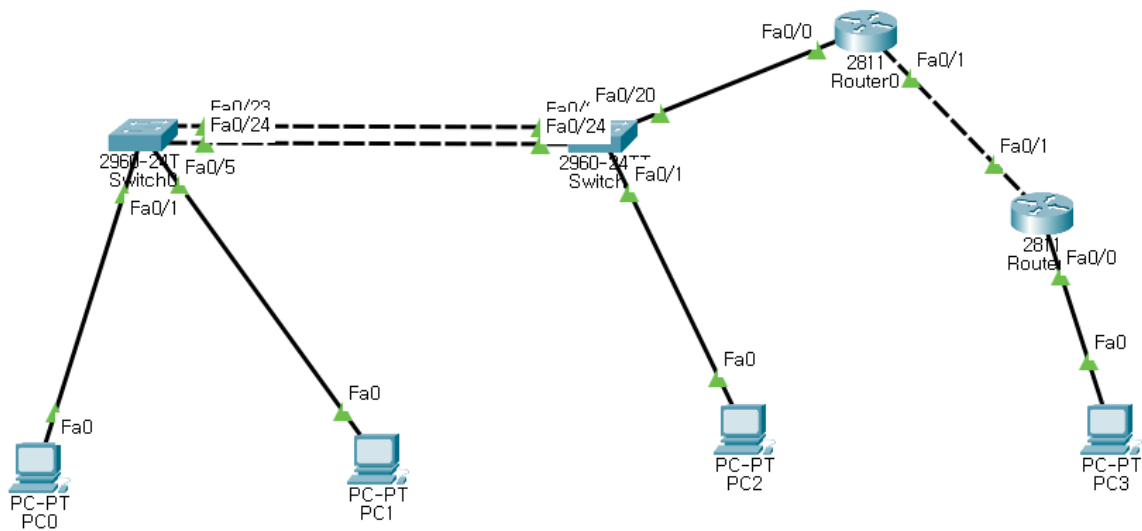
3. 라우터간 네트워크 주소는 172.16.20.4/30
- 상위 라우터 인터페이스 ip에 작은 값을 사용할 것

Diagram details:
- Switch 1: f0/23, f0/24, f0/20, f0/1, f0/5. VLAN 11 (1.1.1.2/24), VLAN 15 (1.1.1.3/24).
- Switch 2: f0/20, f0/1, f0/5. VLAN 11 (20.20.20.1/24), VLAN 15 (20.20.20.2/24).
- Connection: f0/20 to f0/20. IP addresses: 172.16.20.4/30, 172.16.20.1/24, 20.20.20.1/24.

- 내부통신 가능하게 (router on a stick 방식 사용)
 - 서브인터페이스 번호는 vlan 번호로
- 내부 ↔ 외부 통신 가능하게 (static)

- 스위치 사이 2개의 링크 23,24임
- 스위치간 이더채널 사용하기

정답



```
-- 스위치 0번, 1번 모두 채널 5 그룹으로 묶어주고 스위치 모드 트렁크로
int ran f0/23-24
  channel-group 5 mode active
int port-channel 5
  sw mo trunk
```

```
** LSW
en
conf t
vlan 5
vlan 11
int f0/1
  sw acc vlan 11
```

```

int f0/5
  sw acc vlan 5
int ran f0/23-24
  channel-group 5 mode active
int port-channel 5
  sw mo trunk

```

- vlan 5,11 번 생성해주기
- f0/1 인터페이스는 vlan 11번 설정 (1.1.1.0) 대역
- f0/5 인터페이스는 vlan 5번 설정 (5.5.5.0) 대역
- 스위치 0번에 f0/23-24 인터페이스 묶어서 접근
 - 채널 5 그룹으로
- 스위치 0번 이더넷채널 5번에 접근
 - 스위치 모드 트렁크로 활성화

```

** RSW
en
conf t
vlan 11
vlan 5
int f0/1
  sw acc vlan 11
int ran f0/23-24
  channel-group 5 mode active
int port-channel 5
  sw mo trunk
int f0/20
  sw mo trunk

```

- 마찬가지로 vlan 11번, 5번 생성
- 인터페이스 f0/1 vlan 11번으로 스위치 액세스
- 마찬가지로 f0/23-24 이더넷 채널 5로 활성화
- 인터페이스 채널5로 접근 후 스위치 모드 트렁크로
- f0/20 인터페이스

- 라우터랑 연결되어 있기에 스위치 모드 트렁크로 vlan 트래픽을 구분해야함으로.

```

*** R1
en
conf t
int f0/0.11
    encap dot1q 11
    ip add 1.1.1.1 255.255.255.0
int f0/0.5
    en dot 5
    ip add 5.5.5.1 255.255.255.0
int f0/0
    no shut
!
ip route 20.20.20.0 255.255.255.0 172.16.20.6

```

- 라우터 온 어 스틱
 - 스위치와 라우터를 연결할 때 2개의 vlan 트래픽을 구분하기 위한 가상 게이트웨이 생성
 - 인터페이스 f0/0 번에 0.11 즉 11번 vlan 생성, 0.5 5번 vlan 생성
 - 그리고 각각 게이트웨이 등록
 - 인캡슐레이션 vlan 5,11 / dot1q에 vlan 5,11 등록
- f0/0 no shut 으로 up 해줘야함
- 라우터 설정 라우터1은 왼쪽(1.1.1.0, 5.5.5.0 대역은 알지만 20.20.20.0 대역은 모름)
 - ip route 20.20.20.0 대역 등록하고 넥스트홉으로 반대편 라우터 인터페이스 선택 (172.16.20.6 또는 f0/1)

```

*** R2
en
conf t
ip route 5.5.5.0 255.255.255.0 172.16.20.5
ip route 1.1.1.0 255.255.255.0 172.16.20.5

```

- R2 라우터는 20.20.20.0 대역은 알고 있지만 반대편은 모르기에 라우팅 설정을 해준다.
- 마찬가지로 next hope으로 반대편 인터페이스의 ip를 지정 또는 f0/1