

6/24

네트워크

cloud 필요 네트워크 기반으로 4일 수업 예정 3계층이다..

cisco packet tracer 시뮬 / gns 3 에뮬 실습환경

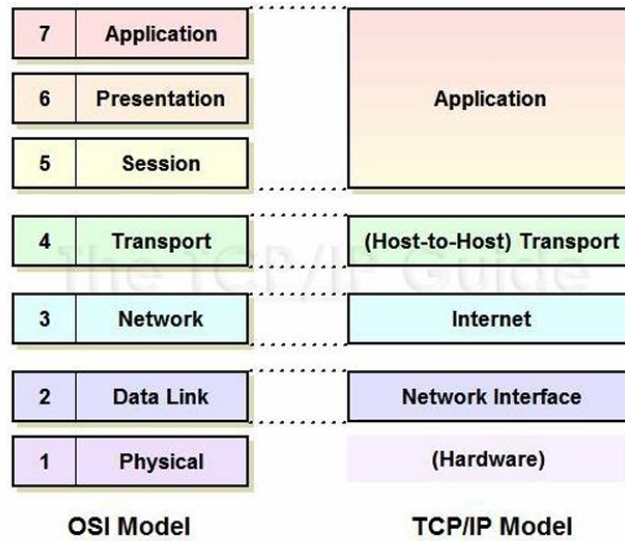
쓰기 편한건 packet tracer

gns3는 ios image가 있어야 사용가능

1. gns3에서 c3660 라우터로 에뮬레이터 실행
이 에뮬레이터로 구동한 라우터는 putty와 같이 특정 어플위에서만 디버깅? 가능
해당 실습에선 mobasxterm으로 진행
2. mobasxterm 실행

aws에서 vpc 배운다.

- LAN
SWITCH라는 2계층 장치로 단말들 연결한 상태
회사에서 뭔갈 구축한다? 다 LAN이다.
- WAN
위의 LAN들을 연결한 상태
- INTERNET
전체 네트워크를 연결시켜 놓은 상태
- PROTOCOL
어떤 장비든 표준화된 PROTOCOL로 만든 장비면 서로 간 통신이 된다.
- OSI 7계층

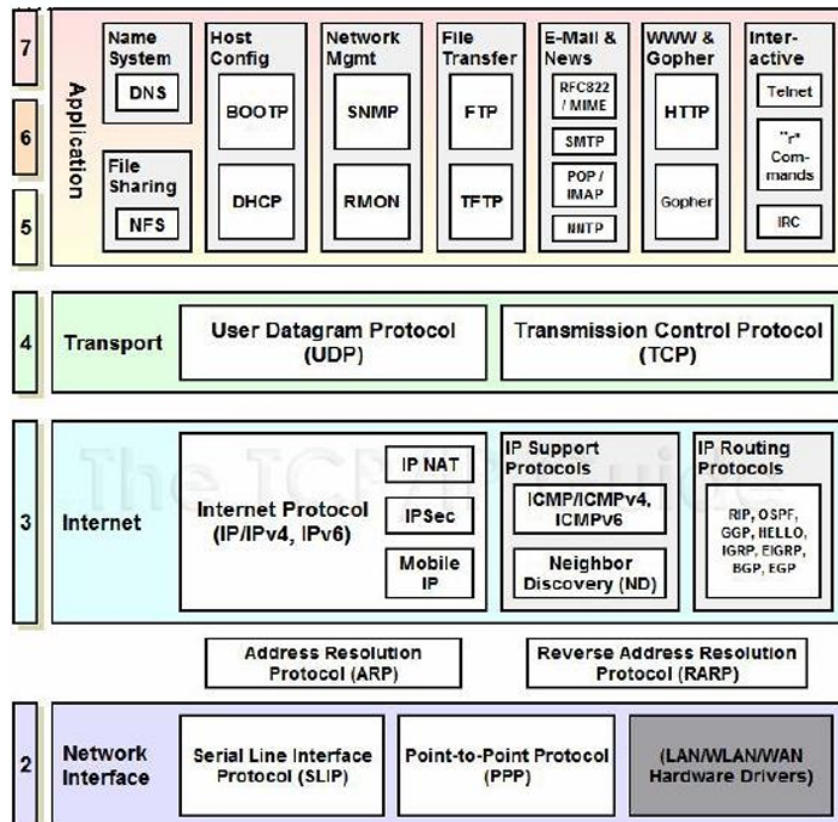


계층별 기능들 데이터가 어떻게 만들어지고 매체를 통해 나가는가.

계층화를 해놓으면 장비나 프로그램을 만들 때 용이하다. 자기 계층의 일만 하면 되니까

오른쪽은 실제 동작하는 프로토콜을 분류해 놓은 것 흡사하다.

- TCP IP 대표 프로토콜

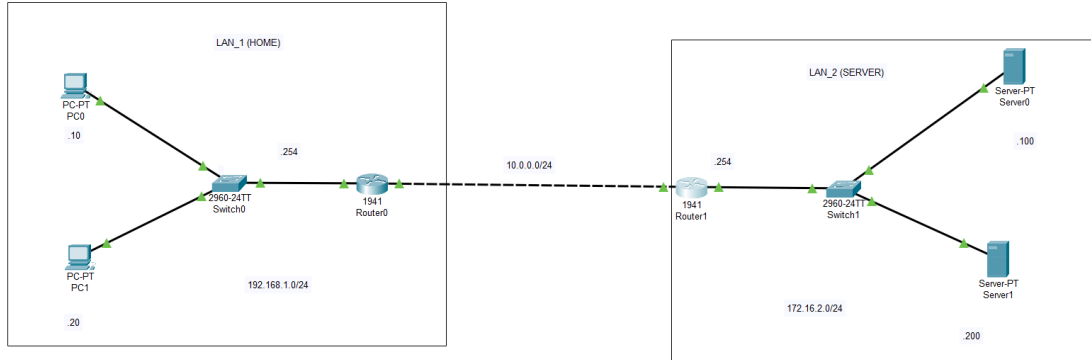


스위치는 2계층까지만 읽는다.

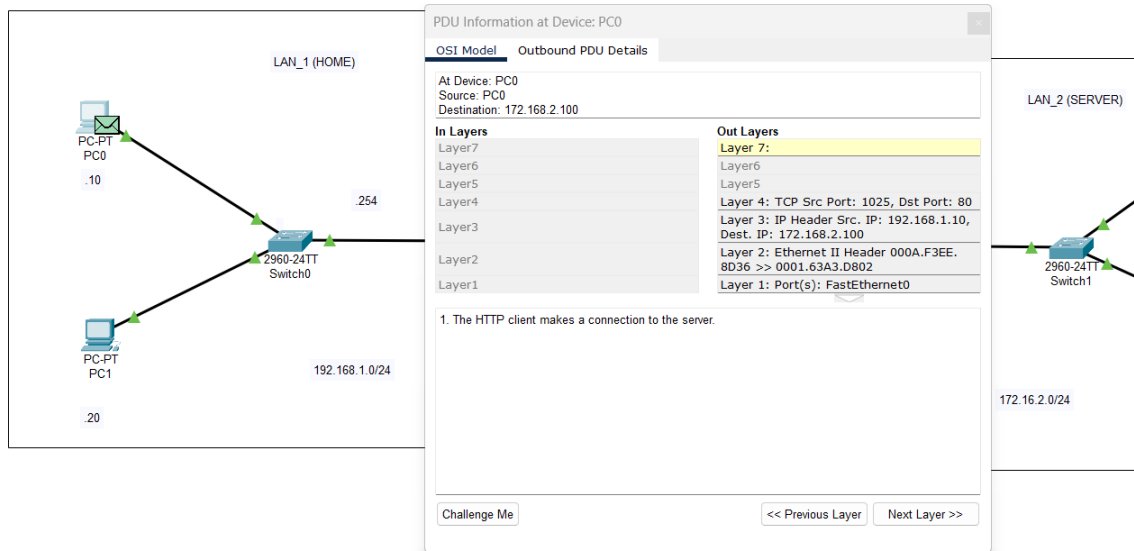
라우터는 3계

라우터에서 목적지에 대한 정보가 없으면 버린다.

목적지에 대한 정보를 네트워크 엔지니어가 넣어줘야



```
Router(config)#exit
Router(config)#interface GigabitEthernet0/1
Router(config-if)#
Router(config-if)#exit
Router(config)#interface GigabitEthernet0/1
Router(config-if)#
Router(config-if)#exit
Router(config)#ip route 192.168.1.0 255.255.255.0 10.0.0.1
Router(config)#
```

목적지까지 가는데에는 3계층까지만 필요 도착해선 PORT를 구분해서 적절한 SERVICE에게 전달을 해준다. HTTP를 요청했을 시 헤더의 정보는 위와 같다.

스위치는 MAC 주소로 통신을 한다.

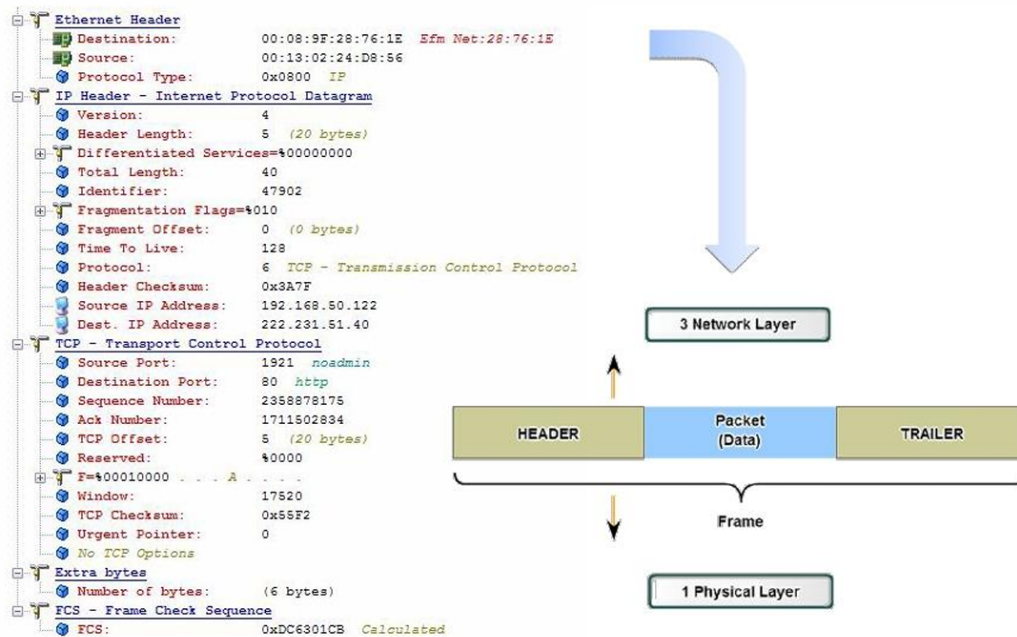
상대방과의 통신을 위해선 상대방의 주소를 알아야 한다. 상대방을 식별하기 위한 2계층의 주소가 MAC이고 MAC을 알아오는 PROTOCOL이 ARP이다.

ARP가 동작을 못하면 통신할 수가 없다.

V4 → V6로 가면서 BROADCAST가 없어진다.

LAN ↔ LAN IP ROUTER

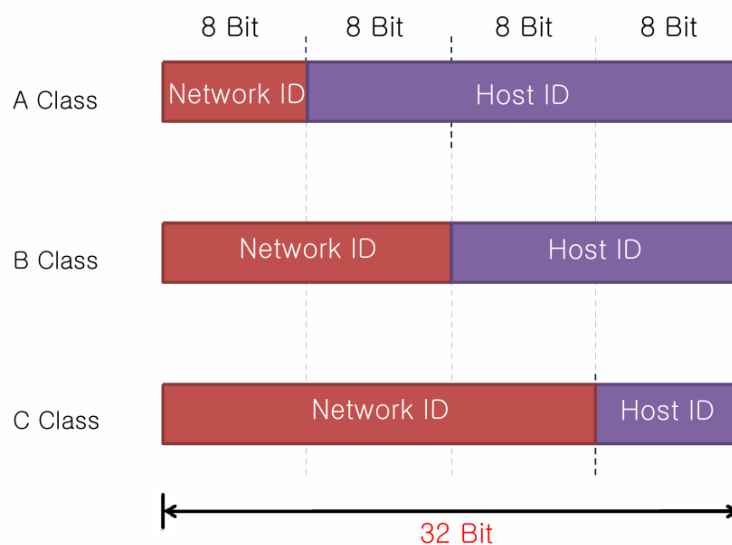
Packet의 구성요소



Frame 내에 계층들의 정보가 담겨있다.

AWS의 cloud infra 쪽에는 ip를 신경써서 할당해야 한다. 여타 서비스를 쓰는 것과 달리

IP 주소의 클래스



IP 주소만 봐서는 네트워크 구분 불가 NET MASK로 구분한다.

NET MASK를 3개의 CLASS A,B,C로 구분해 놓았다.

Class	IP 주소의 첫번째 옥텟	첫째 옥텟의 최소값 (2진수)	첫째 옥텟의 최대값 (2진수)	첫째 옥텟의 값의 범위 (10진수)	이론적 IP 주소 범위
A Class	0xxx xxxx	0000 0001	0111 1110	1 ~126	1.0.0.0 ~ 126.255.255.255
B Class	10xx xxxx	1000 0000	1011 1111	128 ~ 191	128.0.0.0 ~ 191.255.255.255
C Class	110x xxxx	1100 0000	1101 1111	192 ~ 223	192.0.0.0 ~ 223.255.255.255
D Class	1110 xxxx	1110 0000	1110 1111	224 ~ 239	224.0.0.0 ~ 239.255.255.255
E Class	1111 xxxx	1111 0000	1111 1111	240 ~ 255	240.0.0.0 ~ 255.255.255.255

A, B, C 만 신경 쓰면 된다.

IPV4 주소 : 32비트 , 8비트씩 10진수로 표현하고 .으로 구분

NETMASK : 연속된 1값과 연속된 0 값으로 구성 연속된 1값까지가 네트워크 주소를 나타냄.

연속된 0값은 호스트 주소를 나타냄

IPV4 주소에서 네트워크 부분을 알려면 NETMASK를 확인해야 함.

Default netmask : a클래스 -255.0.0.0

b클래스 - 255.255.0.0

c클래스 - 255.255.255.0

172.16.100.9 ← 앞의 주소만 보면 b클래스임

172.16까지가 네트워크 주소이다.

마스크가 255.255.0.0이기에

subnet이 달라지면 255.255.255.0이 되면

172.16.50.9는 다른 LAN에 위치하게 된다.

255.0.0.0

A클래스는 255* 255* 255 개의 경우의 수가 존재한다. HOST가 이걸 감당 가능한 장비가 없다..

다 쪼개서 사용

VPC를 SUBNET으로 잘라서 사용

IPV4에선 IP 주소랑 NET MASK가 있어야지 네트워크가 어디까지 인지 알 수 있다.

주어진 네트워크가 B클래스여도 C클래스로 SUBNETING 해서 사용한다.

여기서 공인 주소와 사설 주소 개념이 등장한다.

- 공인 IPV4 주소

인터넷에서 유일한 주소 모든 주소에 공인 주소를 사용하게 되면 인터넷이 필요하지 않은 장치에도 공인주소를 할당해야 한다. ← 공인 주소가 부족해진다.

- 사설 IPV4 주소

중복해서 사용할 수 있는 주소이다. 인터넷이 불가능하다.

A 클래스 - 10.0.0.0

B 클래스 - 172.16.0.0~ 172.31.0.0

C 클래스 192.168.0.0

NAT라는 프로토콜을 사용해서 사설 주소를 공인주소로 변경하여서 사용할 수 있게끔 해준다.

V2C를 만들 때 사설 주소 사용

- CIDR (Classless Inter Domain Routing)

- 클래스 개념을 사용하는 네트워크에서는 netmask를 사용하여 네트워크를 구분함. 이러한 클래스 개념을 사용하지 않고 네트워크 비트 갯수를 이용해서 네트워크를 표현하는 방식. prefix 방식

- /네트워크비트갯수 /16 /24 /25 이런 식으로

- 172.16.0.0 255.255.255.0 172.16.0.0 255.255.254.0

172.16.1.0 255.255.255.0

슈퍼네팅

192.168.0.0 255.255.255.0 192.168.0.0/23

172.16.1.0 255.255.255.0

Class	IP 주소의 첫번째 옥텟	첫째 옥텟의 최소값 (2진수)	첫째 옥텟의 최대값 (2진수)	첫째 옥텟의 값의 범위 (10진수)	이론적 IP 주소 범위
A Class	0xxx xxxx	0000 0001	0111 1110	1 ~126	1.0.0.0 ~ 126.255.255.255
B Class	10xx xxxx	1000 0000	1011 1111	128 ~ 191	128.0.0.0 ~ 191.255.255.255
C Class	110x xxxx	1100 0000	1101 1111	192 ~ 223	192.0.0.0 ~ 223.255.255.255
D Class	1110 xxxx	1110 0000	1110 1111	224 ~ 239	224.0.0.0 ~ 239.255.255.255
E Class	1111 xxxx	1111 0000	1111 1111	240 ~ 255	240.0.0.0 ~ 255.255.255.255

0은 모든 네트워크를 나타내서 빠지고

127은 rootback이다.

b클래스 네트워크를 서브네팅하기

서브네팅은 주어진 넷마스크의 네트워크 비트를 늘려가는 것
네트워크 비트가 늘어나면 네트워크 2^n 승으로 나뉘어진다.

172.16.0.0 /16 네트워크를 4개의 서브넷으로 서브네팅

4개니까 2개만 바꾸면 된다.

10101100.00010000. 00 000000.00000000

00, 01, 10, 11

11111111.11111111. 11 000000.00000000

c클래스로 하겠다

172.16.0.0/16 네트워크를 c클래스 네트워크로 서브네팅

10101100.00010000. 00000000.00000000

~ 2^8 승의 가짓수

11111111.11111111. 11111111.00000000

c클래스 정도로 쪼개도 충분하다

주어진 네트워크를 쪼개서 작은 네트워크로 나눈다.

00000000 ⇒ 네트워크 ID (주소)

1~254

11111111 ⇒ 브로드캐스트주소

호스트에다 PC에다 할당할 수 없다. 첫번째 주소랑 마지막 주소는

VPC에다 만들면 5개 더 빠진다. 예약된 주소가 있다.

■ 슈퍼네팅

172.16.0.0/24, 172.16.1.0/24 2개의 네트워크를 하나의 네트워크로 슈퍼네팅

슈퍼네팅은 주어진 네트워크를 하나의 네트워크로 합치는 것.

라우팅 정보의 양을 줄이기 위해 사용함.

네트워크 비트를 호스트 비트로 바꾸면 됨. 바꾼 비트수의 2^n 의 네트워크가 하나로 합쳐짐

172. 16. 00000000 0 . 0

172.16.00000000 1.0

255.255. 1111111 0. 0 255.255.254.0

네트워크 비트가 하나가 준다. 서로 다른 것의 마스크를 0으로 하면 된다.

서브네팅은 c클래스 정도로 쪼개면 된다.

슈퍼네팅을 할 일이 생긴다

172.31.128.0/24

172.31.129.0/24

172.31.130.0/24

172.31.131.0/24

172.31.100000 00

172.31.100000 01

172.31.100000 10

172.31.100000 11

172.31.111111 00 / 22

172.16.0.0/16 64개의 서브넷으로 서브네팅

~2^6개

172.16.0.0/22

172.16.4.0/22

172.16.8.0/22

172.16.252.0/22

172.16.111111 00.0 /22