둘째 주 네트워크 기초 둘째 날

노트북: 필기노트
만든 날짜: 2019-06-07 오후 5:08 **업데이트:** 2019-06-11 오전 12:42
작성자: 이종민
태그: 2주차, 네트워크 기초

네트워크 기초 2일차

IANA : IP Address, 최상위 도메인, AS 번호 보는 사이트 (특수 ip 주소 목록)

서버에 웹 서비스를 작동하고 싶으면 → 웹 서비스에 대한 서비스 설치 → 윈도우즈는 다른 설정 없이 설치할 때 "예"를 클릭하면 허용 되지만, 리눅스는 프로세스를 설치하고 웹 서비스에 대한 방화벽을 추가를 해줘야한다. 리눅스는 내부망과 외부망을 연결해주는 방화벽이 있다. (운영체제에 방화벽이 있고, 프로그램에도 방화벽이 있다) 기본적으로 사용하는 프로그램에 대한 방화벽은 열 수 있어야한다.

4계층 프로토콜과 포트번호를 지정 해당하는 서비스를 프로토콜이 어떤 포트를 사용하는지에 대하여 알아야한다.

윈도우는 라이센스가 있어서 방화벽 (유료)로 제공.

OSI 7 Layer

- L7 Application : 해당하는 서비스를 사용하기 위한 클라이언트와 서버
    - 장비 : L7 스위치 : 웹 방화벽 (웹에 대한 방화벽)
- L6 Presentation : 데이터를 어떻게 표현해서 전송할 것이냐 (인코딩 방식, 압축, 암호화)
    - 장비 : L6 스위치 : 인코딩을 위한 스위치
- L5 Session : 세션을 맺고, 연결하고, 유지하고, 정리한다.
    - 특정한 어플리케이션을 다루기 위해서 (L5, L6) 스위치도 있지만, 해당 서비스에 대한 솔루션을 한다. / 최근엔 웹에 대한 서비스가 많다보니 L7 스위치가 필수로 사용이 된다.
    - 5~7 계층 프로토콜 :FTP, SSL, SSH, SMTP, IMTP, DHCP, POP3, HTTPS 등등
    - 5~7 레이어는 하나로 묶어서 서비스라고 지정한다.
- L4 Transport : Tcp (3hand shake), (4hand shake)를 해서 데이터 전송에 대한 확인을 받는다. / Udp 보내는쪽에서 일방적으로 데이터를 전송을 한다.
    - 장비 : L4 스위치 : 1~4 계층에 대한 일을 할 수 있다. IP와 포트번호, 세션번호를 보고 변환한다.
    - 프로토콜 : TCP, UDP
    - PDU : datagram(udp), segment(tcp)
    - L4 스위치 이상을 붙인다는 것은 로드 밸런싱 목적으로 설치를 한다. (서버가 부화 걸리지 않게 하기 위해서) IP와 포트 번호를 보고 같은 요청인지 다른 요청인지 구분을 한다. 로드 밸런싱 목적으로 앞뒤로 L4 스위치를 붙여서 데이터가 나가기도 하고 들어오기도 한다.
- L3 Network : 주소를 어떻게 찾아갈거냐, 길 찾기 (라우팅)
    - 장비 : Router, L3 스위치

- 프로토콜 : ICMP(IP 에러를 알려줌), IGMP(그룹통신), ARP(IP에 해당하는 맥 주소를 알아오는 것), RARP (맥 주소에 해당하는 IP를 알아오는 것), IP
- PDU : Packet

라우터는 스위치와 다르게 스스로 학습을 하지 않으므로 IP로 된 라우팅 테이블을 엔지니어가 설정한다. 모르는 IP 주소는 버린다.

동적 라우팅 (다이나믹) : 변화하는 상황에 맞춰 경로를 재설정하는 방식
정적 라우팅 (스테틱) : 엔지니어가 경로를 지정하는 방식

- L2 Datalink : 보내는 사람과 받는 사람 MAC 보고 주소로 전송을 하는 것
  - 장비 : Switch
  - 프로토콜 : 이더넷
  - PDU : Frame
  - 특징 : 스위치는 스스로 학습하고 해당하는 포트로만 포워딩, 필터링, 데이터를 모르면 브로드캐스트로 물어보고, 데이터가 오래되면 에이징으로 삭제한다.
- L1 Physical : 컴퓨터에 대한 0,1 의 값을 케이블을 타고 바꾸는 것 신호를 받으면 물리적, 전기적 신호로 바꿔서 전송
  - 장비 : 허브 : (플러딩)으로 브로드 캐스트 뿌림
  - PDU : Bit

MAC 주소

피지컬 어드레스 : mac 주소, 이더넷 안으로 들어간다. 물리적인 어뎁터 안에 제조에서 표현이 된다.
16 진수 = 6byte, 앞에 3 자리는 벤더 정보 / 48 byte 제조사 24byte, 제품번호 24byte

서브넷마스크

IPv4 : 8비트씩 표기 (첫 번째 숫자 옥탭에 따라 클래스를 나눈다)

- D클래스는 서브넷 마스크 크기가 정해져있다 (멀티캐스트)
- E클래스는 연구 목적으로 쓰기 위한 클래스

서브넷팅을 하는 이유 : 공인 IP가 부족하니 아꾸어서 쪼개서 쓰려고 했다.
네트워크가 크면 클수록 PC가 많을수록 브로드캐스트가 많아진다. 그래서 현재는 사설 IP를 사용해서 공인 IP가 부족하기 않기에 브로드캐스트를 줄이기 위해서 서브넷팅한다.

DNS Service(domain name service) : 예전에는 컴퓨터랑 IP마다 이름을 적어놨다. HOST 파일에 각자 알고 있는 파일을 참고해서 만들었고, 서버가 추가 될 때마다 HOST 파일에 IP랑 도메인 네임을 추가 시켰다. (도메인이 너무 많기에 트리 구조로 만들었다.)

특수 IP 주소 목록
10.0.0.0 ~ 10.255.255.255 : 클래스 A에 속하는 사설 IP 주소
172.16.0.0 ~ 172.31.255.255 : 클래스 B에 속하는 사설 IP 주소
192.168.0.0 ~ 192.168.255.255 : 클래스 C에 속하는 사설 IP 주소

클라이언트 : 서비스 요청자
서버 : 서비스 요청자에 대한 응답자
서버 특징 : 클라이언트 요청을 처리하기 위해 항상 대기 상태 / 서버 대기상태 (리스닝)
여러 클라이언트가 동시에 요청하는 경우 안정적으로 처리해야함 (무한대가 아니다)

프로토콜

FTP(20,21) : 파일 전송을 위한 프로토콜, 비암호화 방식이라 현재는 사용을 안한다. | 데이터를 주고 받을 땐 20번 포트, 로그인을 할 때 21번 포트로 통신한다.

피드백 : 웹 호스팅 업체에서 사용, (SSH, SSL을 사용해 암호화를 해서 사용함)
윈도우에선 IIS / 리눅스 SSH을 이용한 FTP, 패시브모드(수동적), 액티브모드(적극적) 등이 있음.

SSH(22) : 원격제어 프로토콜 암호화 기법을 사용한다.
Telnet(23) : 원격제어 프로토콜 / 비암호화 방식이라 현재는 사용하지 않고 (SCP, SSH나 VPN을 사용해서 접속한다)
SMTP(25) : 메일 서비스
DNS(53) : 도메인 이름을 IP 주소로 변환하는 프로토콜
DHCP(UDP 포트 67, 68) : IP 주소를 자동으로 할당하는 프로토콜
TFTP(69) : 파일 전송 프로토콜 (UDP 방식으로 사용자가 직접 에러를 확인해야됨) / 사용하지 않는다.
HTTP(80) : 인터넷을 위해 사용하는 가장 기본적인 프로토콜 / 현재는 HTTPS(443) 사용
POP3(110) : 메일 서비스 프로토콜 (전송된 메일 확인, 예 : outlook)
NetBIOS(138) : 윈도우 시스템 간 대화, 통신 프로토콜
IMTP(143) : 메일 서비스 프로토콜 (메일을 읽은 후 서버에 남음, 예 : NAVER, DAUM)
피드백 : Ping에도 쓰이는 프로토콜
SNMP(161) : 멀리 떨어진 네트워크 장비를 원격으로 제어
SYSLOG(514) : 시스템 로그를 한쪽에 모은다.
snmp : 네트워크 엔지니어 사용 / syslog : 서버 엔지니어 사용
HTTPS(웹암호화) = SSL = TLS

HTTP 서비스
GET : URL의 끝인 파라미터 부분으로 주소로 넘긴다. (주소에서 보낸다)
POST : URL에 파라미터에 정보가 표시되지 않고 인코딩 하여 서버 전송
- 예 : method = get인 값은 주소창에 정보가 노출되지만 post인 값은 서버로 전송한다.
맹점인게 HTTP에서는 패킷에서 post 값을 볼 수 있다. 그래서 암호화인 HTTPS를 사용한다.

HTTP response code 종류
1XX 정보 전송 단순 정보를 제공함
2XX 요청 처리 성공 요청이 성공적으로 이루어짐
3XX 리다이렉션 요청한 해당 자원이 다른 곳에 있음
4XX 클라이언트 에러 요청(클라이언트)에 문제가 있음
5XX 서버에러 서버에 에러가 있음
HTTP 1.0 / 1.1 차이
1.0 : 클릭 할 때 마다 세션을 맺고, 갖고 온다.
1.1 : 세션을 일정시간 동안 유지

예 : 패킷 트레이서에서 http 패킷을 성공적으로 전송하면 200으로 표기됨