# 2회차 과제 제출[네트워크기초,TCP/IP 4계층, 네트워크 기기, IP주소, HTTP]

## 2.1 네크워크 기초

네트워크란 노드(서버, 라우터, 스위치)와 링크(유선, 무선)가 연결되어 리소스를 공유하는 집합.

즉 가상의 공간에서 데이터를 공유함.

### 2.1.1 처리량과 지연시간

좋은 네트워크란 많은 양을 처리할 수 있으며 지연시간이 짧고 좋은 보안을 갖춘 네트워크.

처리량(throughput) : 링크를 통해 전달되는 단위 시간당 데이터양.

처리량은 트래픽, 네트워크 장치간의 대역폭, 네트워크 중간에 발생하는 에러, 장치의 하드웨어 스펙에 영향을

받는다.

지연시간(latency) : 요청이 처리되는 시간. 특정 메세기자 두 장치 사이를 왕복하는데 걸리는 시간.

지연시간은 매체타입, 패킷크기, 라우터의 패킥 처리 시간에 영향을 받는다.

### 2.1.2 네트워크 토폴로지와 병목 현상

네트워크 토폴로지란 노드와 링크가 어떻게 배치되어 있는지에 대한 방식이자 연결 행태를 의미한다.

1. 트리 토폴로지(tree topology)

계층 토폴로지라고 하며 트리형태로 배치한 네트워크 구성.

2. 버스 토폴로지(bus topology)

중앙 통신회성 하나에 여러개의 노드가 연결되어 공유하는 네트워크 구성. (근거리(LAN) 전용)

- 설치비용이 적고 신뢰성이 우수하며 중앙회선에 노드 추가 및 삭제가 수월함.
- 스푸핑이 가능하다는 문제점이 있음.(LAN상에서 패킷이 송신과 관련없는 다른 호스트에 가지 않도록 하는 스위칭 기능을 마비시키거나 속여서 특정 노드에 해당 패킷이 오도록 처리하는 것.)

3. 스타 토폴로지(star topology)

중앙에 있는 노드에 모두 연결된 네트워크 모양새.

- 노드 추가, 에러 탐지가 수월함. 패킥 충돌 발생 가능성 적음. 왜여?
- 중앙노드가 아닌 노드에 에러가 발생하면 쉽게 에러를 발생할 수 있으며 장애노드가 중앙 노드가 아닐 경우 타 노드에 영향을 끼치는 것이 적음.
- 중앙노드에 문제가 생기면 전체적인 네트워크 마비. 설치비용 고가
- 이 형태의 경우 해킹을 한다고 했을 때 가장 쉽게 마비시킬 수 있느 네트워크 형태일 것 같다.

4. 링형 토폴로지(ring topology)

각각의 노드가 양 옆의 두 노드와 연결하여 전체적으로 고리처럼 하나의 연속된 길을 통해 통신하는 망 형태의 네트워크.

5. 메시 토폴로지(mesh topology)

메시 토폴로지는 그물망처럼 연결되어 있는 구조이며 망형 토폴로지라고도 한다.

- 한 단말 장치에 장애가 발생해도 여러 개의 경로가 존재하므로 네트워크를 계속 사용할 수 있고 트래픽 분산 처리 가능.
- 노드의 추가가 어렵고 구축 비용과 운용비용이 고가.

> 네트워크의 구조라고도 말하는 토폴로지가 중요한 이유는 병목현상을 찾을 때 중요한 기준이 된다.

### 2.1.3 네트워크 분류

네트워크는 규모를 기반으로 구분한다.

LAN(Local Area Network) :  근거리 통신망. 같은 건물이나 캠퍼스 정도의 공간에서 운영함.

전송속도 빠름, 혼잡도 낮음.

MAN(Metropolitan Area Network) : 대도시 지역 네트워크. 도시 정도의 넓은 지역에서 운영함.

전송속도 보통,혼잡도 보통.

WAN(Wide Area NetWork) : 광역 네트워크. 국가나 대륙 정도의 광활한 지역에서 운영함.

전송속도 낮음, 혼잡도 높음.

병목(bottleneck) 현상 : 전체시스템의 성능이나 용량이 하나의 구성요소로 인해 제한을 받는 현상. 서비스에서 이벤트를 열었을 때 트래픽이 많이 생기고 트래픽을 잘 관리하지 못하면 병목현상이 생겨 사용자는 웹사이트로 들어가지 못한다.

네트워크 병목 현상의 주된 원인

1. 네트워크 대역폭
2. 네트워크 토폴로지
3. 서버 CPU, 메모리 사용량
4. 비효율적인 네트워크 구성

**병목현상이 의심되는 상황**일 때에는 네트워크와 관련된 테스트, 무관한 테스트를 통해 네트워크 부분에서 발생한 문제임을 확인한 후 네트워크 성능 분석을 진행해야 함.

2.1.4 네트워크 성능 분석 명령어

1. ping : 네트워크 상태를 확인하려는 대상 노드를 향해 일정 크기의 패킷을 전송하는 명령어.

   연결이 잘 되어 있는지, 전송시간이 어느정도 되는지 확인함.

2. netstat : 접속되어 있는 서비스들의 네트워크 상태를 표시하는 명령어.

   서비스 포트가 열려있는지 확인함.

3. nslookup : DNS에 관련된 내용을 확인하기 위한 명령어.

   특정 도메인에 매핑된 ip를 확인하기 위해 사용함.

4. tracert : 목적지 노드까지 네트워크 경로를 확인할 때 사용하는 명령어.

   목적지 노드까지 구간들 중 어느 구간에서 응답 시간이 느려지는 지 등을 확인할 수 있음.

2.1.5 네트워크 프로토콜 표준화

> 네크워크 프로토콜이란 다른 장치들끼리 데이터를 주고받기 위해 설정된 공통된 인터페이스.

약소된 프로토콜이 있기 때문에 데이터를 서로 주고 받을 수 있음.

# 2.2 TCP/IP 4계층 모델

인터넷 프로토콜 스위트는 인터넷에서 컴퓨터들이 서로 정보를 주고 받는 데 쓰이는 프로토콜의 집합이며,

이를 TCP/IP 4계층 혹은 OSI 7계층으로 설명한다.

## 2.2.1 계층 구조

TCP/IP 계층은 4개의 계층을 가지고 있으며 각 계층에 문제가 발생하거나 변동사항이 있더라도 다른 계층에 영향을 주지 않는다.

1. 애플리케이션 계층(appilcation)

실질적으로 사람들에게 서비스를 제공하는 단계

응용프로그램 :  FTP, HTTP, SSH, SMTP, DNS

2. 전송 계층(transport)

송신자와 수신자를 연결하는 통신서비스.

핵심 기능 : 연결지향 데이터 스트림 지원, 신뢰성, 흐름제어, 데이터 전달

전송계층의 데이터 전달 방식 2가지

- TCP : “가상회선 패킷 교환 방식”
  - 수신여부를 확인함.
  - 패킷 사이의 순서를 보장하고 연결 지향  프로토콜을 사용해서 연결을 하여 신뢰성을 구축함.
  - 신뢰성 확보 시에 3-웨이 핸드쉐이크 라는 작업을 한다.
- UDP :  “데이터그램 패킷 교환 방식”
  - 수신여부를 확인하지 않음
  - 패킷의 순서를 보장하지 않고 데이터만 주고 받는 형태

가상회선 패킷 교환 방식 : 각 패킷에 가상회선 식별자가 있고 모든 패킷의 전송이 완료되면 가사회선이 해체되고 패킷들은 전송된 순서대로 도착함.

데이터그램 패킷 교환 방식 : 패킷이 최적의 경로를 선택하여 자주적으로 이동한다.

하나의 메시지에서 분할된 여러 패킷은 서로 다른 경로로 전송될 수 있으므로 출발 순서와 도착한 순서가 다를 수 있다.(예시.123 ⇒ 213)

TCP 연결 과정

3-웨이 핸드쉐이크 : 클라이언트와 서버가 통신하는 방법 중 하나

- 이러한 과정으로 신뢰성이 구축되고 데이터 전송을 시작한다.
- TCP는 위처럼 신뢰성 구축의 과정이 있고 UDP는 없다.

1. SYN 단계 : 클라이언트가 서버에 ISN과 SYN을 보낸다.
2. SYN+ACK 단계 : 서버는 클라이언트의 SYN을 수신하고 서버의 ISN과 승인번호(클라이언트의 ISN+1)를 보낸다.
3. ACK 단계 : 클라이언트는 ACK 와 승인번호(서버의 ISN +1한 값)를 서버에 보낸다.

TCP 연결 해체 과정

4-웨이 핸드쉐이크

1. 클라이언트가 연결을 닫으려고 할 때 FIN으로 설정된 세그먼트를 보냄. 클라이언트는 FIN_WAIT_1 상태
2. 서버는 클라이언트로 부터 ACK라는 승인 세그먼트를 보냄. 서버는 CLOSE_WAIT 상태.
3. 클라이언트가 세그먼트를 받으면 FIN_WAIT_2 상태가 됨
4. 서버는 SCK를 보내고 일정 시간 후에 클라이언트에 FIN이라는 세그먼트를 보냄
5. 클라이언트는 TIME_WAIT 상태가 되고 다시 서버로 ACK를 보내서 서버는 CLOSED 상태가 된다.
6. 클라이언트가 일정 시간 대기 후 연결이 닫히고 서버와 클라이언트의 연결이 닫힘.

3. 인터넷 계층(internet)

인터넷 계층은 장치로부터 받은 네트워크 패킷을 IP주소로 지정된 목적지로 전송하기 위해 사용되는 계층.

인터넷 계층 장비 : IP, ARP, ICMP 등

패킷을 수신해야 할 상대의 주소를 지정하여 데이터를 전달.

상대방의 수신확인은 하지 않은 비연결형성을 가짐.

4. 링크 계층(link)

링크계층은 전선, 광섬유, 무선 등으로 실질적으로 데이터를 전달하며 장치간에 신호를 주고받는 ‘규칙’을 정하는 계층.

네트워크 접근 계층이라고도 한다.

전 이중화 통신(full duplex) : 양쪽 장치가 동시에 송수신할 수 있는 방식.

이는 송신로와 수신로로 나눠서 데이터를 주고받으며 현대의 고속 이더넷은 이런 방식을 기반으로 통신함.

CSMA / CD : 이전방식. 유선 LAN에 반이중화 통신.

수신과 송신이 한 경로에서 발생함.

데이터를 보낸 이후 충돌이 발생한다면 일정시간 이후 재전송함.

유선LAN을 이루는 케이블 : TP 케이블.

- 트위스트 페어 케이블 : 구리로 만들어짐
- 광섬유 케이블 : 광섬유로 만들어짐. 레이저를 이용해서 통신하기 때문에 구리선과는 비교할 수 없을 정도의 장거리, 고속 통신이 가능함.(내부에서 계속 반사하여 가는 원리)

## 2.3 네트워크 기기

네트워크 기기는 계층별로 나누어볼 수 있음.

상위 계층을 처리하는 기기는 하위계층을 처리할 수 있지만 하위계층을 처리하는 기기는 상위계층의 기능을 수행할 수 없음.

### 2.3.2 애플리케이션 계층을 처리하는 기기

L7 스위치 : 스위치는 여러장비를 연결하고 데이터 통신을 중재하며 목적지가 연결된 포트로만 전기신호를 보내서 데이터를 전송하는 통신 네트워크 장비이다.

L7스위치는 서버의 부하를 분산시키는 기능을 가져서 ‘로드밸런서’(또다른 로드밸러서로 전송계층의 L4스위치도 있음)라고도 한다.

클라이언트로부터 오는 요청들을 뒤쪽의 여러 서버로 나누는 역할을 하며 시스템이 처리할 수 있는 트래픽 증가를 목표로 한다.

URL, 서버, 캐시, 쿠키들을 기반으로 트래픽을 분산함.

바이러스, 불필요한 외부데이터 등을 필터링함

가지고 있으며 응용 프로그램 수준의 트래픽 모니터링 가능.

장애가 발생한 서버가 있으면 이를 트래픽 분산 대상에서 제외해야 하는데 이를 정기적으로 헬스체크(health check)를 통해서 감시한다.

**L7스위치와 L4스위치의 차이**

- L4스위치는 전송계층을 처리하는 기기로 스트리밍 관련 서비스에서는 사용할 수 없음.
- 메세지를 기반으로 인식하지 못하고 IP와 포트를 기반으로(특히 포트) 트래픽을 분산함.⇒L7의 경우에는 트래픽을 분산할 때 IP, 포트, 외에도 URL, HTTP 헤더, 쿠키 등 고려하는 사항이 더 많다.

ALB(application load balancer) 컴포넌트 : 클라우드 서비스에서 L7을 이용한 로드밸런싱

NLB(Network load balancer) 컴포넌트 : 클라우드 서비스에서 L4를 이용한 로드밸런싱

✅ 헬스체크(health check) : 전송주기, 재전송 횟수 등 특정 패턴으로 서버에 반복 요청을 보내는 것.  L7, L4 모두 헬스 체크를 통해 정상적인 서버인지 비정상적인 서버인지 판별함.

**로드밸런서를 이용한 서버 이중화**

로드밸런서의 대표적인 기능은 서버 이중화이다.

서비스를 안정적으로 운용하기 위해서는 2대 이상의 서버는 필수이다.(하나에 문제가 생겨도 서비스를 유지할 수 있도록)

로드 밸런서는 2대 이상의 서버를 기반으로 가상IP를 할당함.

## 2.3.3 인터넷계층을 처리하는 기기

1. 라우터(router) : 여러 개의 네트워크를 연결, 분할, 구분시켜주는 역할.

**라우팅** = 다른 데트워크에 존재하는 여러 장치끼리 서로 데이터를 주고 받을 때 패킷 소모를 최소화하고 경로를 최적화하여 최소 경로로 패킥을 포워딩함.

2. L3 스위치 : L2기능과 라우팅 기능을 갖춘 장비.

L3를 라우터로 부르기도 한다. 라우터는 소프트웨어 기반의 라우팅과 하드웨어 기반의 라우팅을 하는 것으로 나눈다. 이 중 하드웨어 기반의 라우팅을 담당하는 기기가 L3 스위치이다.

> **MAC**주소 : 각 LAN 카드에 주민등록번호처럼 각각을 구분하기 위해 있는 고유의 식별 번호.

## 2.3.4 데이터 링크 계층을 처리하는 기기

**L2 스위치 : 연결된 장치로부터 패킷이 왔을 때 패킷을 전송함.(패킷의 MAC주소를 읽어서 스위칭함.)**

- ㅁ 목적지가 MAC테이블에 없다면 전체 포트에 전송하고 MAC주소 테이블의 주소는 일정기간 이후 삭제가능
- ㅁ IP주소를 기반으로 라우팅을 불가능

**브리치(bridge) : 두 개의 근거리 통신망을 상호 접속할 수 있도록 하는 통신망 연결장치.**

- ㅁ 포트와 포트 사이의 다리역할을 함.
- ㅁ 장치에서 받아온 MAC주소를 MAC주소 테이블로 관리함.
- ㅁ 통신망 범위를 확장하고 서로 다른 LAN등으로 이루어진 하나의 통신망을 구축할 수 있음.

## 2.3.5 물리계층을 처리하는 기기

**NIC(네트워크 인터페이스 카드) : 2대 이상의 컴퓨터 네트워크를 구성할 수 있음.**

네트워크와 빠른 속도로 데이터를 송수신할 수 있도록 컴퓨터 내에 설치하는 확장 기기

**리피터(repeater) : 들어오는 약해진 신호 정도를 증폭하여 다른 쪽으로 전달하는 장치(더멀리).**

광케이블 보급 이후 잘 쓰지 않는 장비

**AP(Access Point) : 패킷을 복사하는 기기**

AP에 유선 LAN을 연결하면 다른 장치에서 무선 LAN 기술을 사용하여 무선 네트워크 연결 가능

# 2.4 IP주소

## 2.4.1 ARP

컴퓨터와 컴퓨터간의 통신은 IP주소에서 ARP를 통해 MAC주소를 찾아 MAC주소를 기반으로 통신함.

ARP : IP주소로부터 MAC주소를 구하는 IP와 MAC주소의 다리 역할을 하는 프로토콜.

ARP를 통해 가상의 주소인 IP주소를 실제 주소인 MAC주소로 변환함.(반대도 가능)

2.4.2 홉바이홉 통신

**홉바이홉 통신** : IP주소를 통해 통신하는 과정.

⇒ 수많은 서브네트워크 안에 있는 라우터의 라우팅 테이블IP를 기바으로 패킷을 전달하고 또 전달해가며 라우팅을 수행하며 최종 목적지까지 패킷을 전달하는 통신기법.

**라우팅** : IP주소를 찾아가는 과정

홉(hop) : 통신망에서 각 패킷이 여러 개의 라우터를 건너가는 모습을 형상화함.

**라우팅 테이블(routing table)** : 라우터에 들어가 있는 목적지 정보들과 그 목적지로 가기 위한 방법들이 적혀 있는 리스트.

⇒ 송신지에서 수신지까지 도달하기 위해 사용함.

**게이트웨이** : 서로 다른 통신망, 프로토콜을 사용하는 네트워크 간의 통신을 가능하게 하는 관문 역할을 하는 기기를 두루 일컫는 말.

### 2.4.3 IP 주소 체계

IP주소는  IPv4와 IPv6으로 나눈다.

DHCP : IP주소 및 기타 통신 매개 변수를 자동으로 할당하기 위한 네트워크 관리 프로토콜.

- 네트워크 장치의 IP 주소를 수동으로 설정할 필요 없이 인터넷에 접속할 때 마다 자동으로 IP주소를 할당할 수 있음.

NAT : 패킷이 라우팅 장치를 통해 전송되는 동안 패킷의 IP주소 정보를 수정하여 IP주소를 다른 주소로 매핑하는 방법.

- 사설IP를 공인 IP로 변환하거나 공인 IP를 사설 IP로 변환함.
- 공유기와 NAT : NAT을 쓰는 이유는 여러 대의 호스트가 하나의 공인IP를 용하여 인터넷에 접속하기 위한 것.(공유기에 NAT기능이 있어서 여러 기기가  접속할 수 있음.)
- NAT 장점 : 네부 네트워크에 대한 보안력
- NAT 단점 : 접속하는 호스트에 따라 인터넷 속도가 느려질 수 있음.

### 2.4.4 IP주소를 이용한 위치 정보

IP주소는 인터넷에서 사용하는 네트워크 주소 =동, 구 까지 위치추적 가능.

## 2.5 HTTP

http는 기본적으로 애플리케이션 계층의 웹서비스 통신에 사용된다.

### 2.5.1 HTTP/1.0

한 연결당 하나의 요청을 처리함⇒ RTT 증가

RTT : 패킷이 목적지에 도달하고  나서 다시 출발지로 오기까지 걸리는 시간(패킷 왕복 시간)

해결법 : 이미지 스플리팅, 코드 압축, 이미지 Base64 인코딩 등

### 2.5.1 HTTP/1.1

매번 TCP연결을 하는 것이 아니라 한 번 연결 후에 여러 개의 파일을 송수신 할 수 있도록 함.

- 헤더에 많은 메타데이터가 있고 압축이 되지 않아 무거움.

HOL Blocking : 네트워크에 같은 큐에 있는 패킷이 그 첫번 째 패킷에 의해 지연될 떄 발생하는 성능 저하 현상

### 2.5.1 HTTP/2

HTTP/1.x보다  지연시간을 줄이고 응답 시간을 더 빠르게 할 수 있으며 멀티 플렉싱, 헤더 압축, 서버 푸시, 요청의 우선순위 처리를 지원하는 프로토콜

### 2.5.1 HTTPS

**HTTPS** :  애플리케이션 계층과 전송계층 사이에 신뢰계층인 SSL/TLS계층을 넣은 신뢰할 수 있는 HTTP 요청

⇒ 통신을 암호화 함.

- HTTP/2는 HTTPS 위에서 동작함.

SSL/TLS : 클라이언트가 서버와 통신할 때 제 3자가 메세지를 도청하거나 변조하지 못하도록 함.

전송계층에서 보안을 제공하는 프로토콜.

시간관계상 이후 3페이지 정도 못했숨다..ㅜㅜ 내일안에 채워서 재업로드 할게요. 죄송합니다ㅜㅜ!