

TCP/IP

솔직히 나만 CS 지식 업는거 아니자나요

TCP/IP 왜 선택했을까 (후회하는거 아님)

TCP/IP는 프로토콜에서 가장 중요한 역할을 함

현재 대부분 통신이 TCP/IP 환경 기반

그래서 TCP/IP가 무엇일까?

- 가장 최근에 발명된 컴퓨터와 컴퓨터 간의 지역 네트워크 (LAN) 혹은 광역 네트워크 (WAN)에서 원활한 통신을 가능하도록 하기 위한 통신규약(Protocol)
- TCP/IP는 하드웨어, 운영체제, 접속매체에 관계없이 동작할 수 있다는 **개방성**에 있어 인터넷 통신을 위한 핵심으로 채택
- TCP와 IP라는 2개의 프로토콜로 이루어져 있는데, 통상 IP 프로토콜 위에 TCP 프로토콜이 놓여서 TCP/IP라고 부름

IP (Internet Protocol)

- node(단말기)와 node 간의 데이터 패킷을 전송하기 위해서는 각 node에 주소(우편번호와 같은)를 필요로 함
- IP는 4바이트(or 6바이트)로 이루어진 주소번호를 사용하여 각각 node를 구분하고, 목적지를 찾아감 → 이를 IP Address라고 하며, "192.168.100.100"과 같은 형식으로 사용
- 숫자로 된 인터넷 주소를 인간이 식별하기 쉬운 도메인 네임으로 변환시켜주는 Domain Name Service를 사용하여 125.209.222.141를 www.naver.com처럼 만들어 줌

TCP (Transmission Control Protocol)

- 서버와 클라이언트 간에 데이터를 신뢰성있게 전달하기 위해 만들어진 프로토콜
- 데이터는 네트워크 선로를 통해 전달되는 과정에서 손실되거나 순서가 뒤바뀌어 전달 될 수 있는데, TCP는 손실을 검색하여 이를 교정하고 순서를 조합할 수 있게 해줌
- 데이터를 전송하기 전에 데이터 전송을 위한 "연결"을 만드는 연결 지향 프로토콜



host, router는 다양한 종류와 다양한 품질을 가지는 네트워크 회선으로 연결

통화 품질이 다르다? → 데이터를 보내는 중 데이터 손상 가능성 0
→ 데이터의 순서가 바뀔 가능성 0

하나의 고정 경로 사용 X → 여러 임의의 경로 사용
각 경로를 구성하는 회선의 품질이 다르면 도착 시간에 차이 발생

TCP/IP는 연결된 상태에서 올바른 통신을 하도록 도와준다.

근데 엄연히 말하면 TCP가 올바른 통신을 하도록 도와주는 기능을 가지고 있는거고 IP는 TCP 패킷을 전송하는 일만 함

TCP가 가지고 있는 기능

1. 패킷이 빠졌을 경우 재전송을 요청하는 기능
2. 패킷에 일련번호를 줌으로써 서로 다르게 도착될지도 모르는 패킷의 순서를 재조합하는 기능

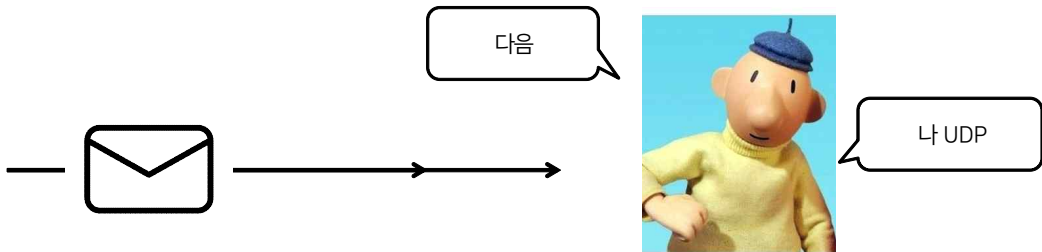


TCP와 UDP

이 둘은 데이터를 전송한다는 점에서 비슷하지만 특성에 있어 차이가 있다.

UDP : User Datagram Protocol

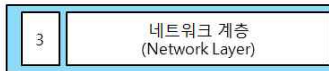
TCP	UDP
연결 지향	비연결지향
자체적 오류 처리, 순서 교정 기능	TCP와 같은 처리 기능 X
데이터의 신뢰도가 중요할 때 사용	단순히 데이터를 주고 받기만 할 때 사용
HTTP, FTP, TELNET 등에서 사용	실시간 멀티미디어 정보 처리 (비디오/오디오 스트레밍)



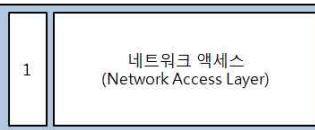
OSI 7계층 맛보기

- Open System Interconnection Reference Model
- OSI는 각종 시스템 간의 연결을 위하여 ISO(국제 표준화 기구)에서 제안
- 시스템에 상관 없이 서로의 시스템이 연결될 수 있도록 만들어주는 모델
- OSI를 계층별로 나눔으로써, 각 계층에서 필요한 부분만을 신경쓰고 개발할 수 있으며 통신 서비스 개발 시간을 줄일 수 있도록 도와줌

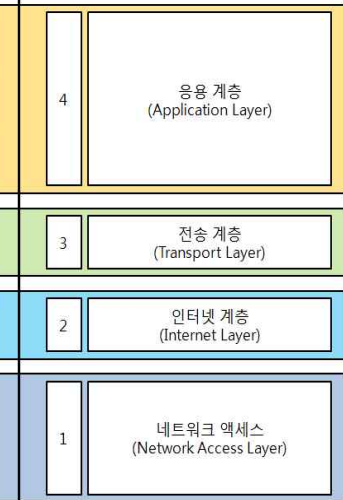
OSI 모형



TCP/IP 4계층



TCP/IP 4계층



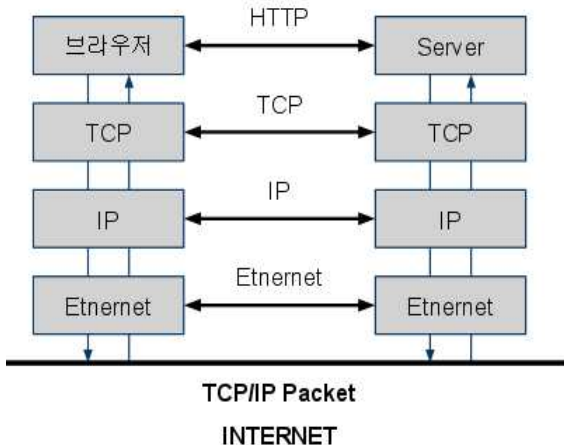
네트워크를 사용하는 응용프로그램(FTP, Telnet, SMTP)등으로 이루어짐
OSI 모델의 Application Layer 와 Presentation Layer, Session Layer를 포함

도착을 원하는 시스템까지 데이터를 전송하기 위한 일을 하는 계층
각각의 시스템을 연결하고 TCP 프로토콜을 이용하여 데이터를 전송
OSI 모델의 Transport Layer를 포함

데이터를 정의하고 데이터의 경로를 지정하는 일(라우팅)을 담당
데이터를 정확히 라우팅하기 위해서 IP 프로토콜을 사용
OSI모델의 Network Layer를 포함

물리적 계층 즉 이더넷 카드와 같은 하드웨어
OSI 모델의 Data Link Layer 와 physical Layer를 포함

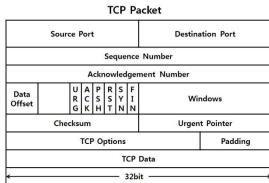
TCP/IP 4계층에 의한 데이터 전송



1. 사용자는 크롬과 같은 브라우저를 사용하여 URL 입력을 통해 웹페이지를 요청한다.



2. 사용자 요청은 인터넷 상에서 전달되기 용이한 패킷으로 만들기 위해서 TCP 패킷으로 만들어진다.



나 TCP 패킷
전달에 편하지



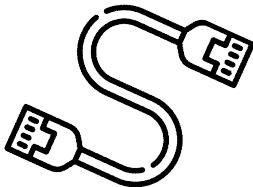
3. 인터넷 상에서 원하는 주소로 이동할 수 있도록 하기 위해 IP 패킷으로 다시 만들어지고 이것은 이더넷 카드
로 보내어져서 Internet 으로 나간다.

TCP Packet

Source Port		Destination Port	
Sequence Number			
Acknowledgement Number			
Data Offset	URG	ACK	PSH
	RST	SYN	FIN
Checksum		Windows	
Urgent Pointer			
TCP Options		Padding	
TCP Data			
32bit			



나 IP 패킷
내 주소, 도착 주소 있음



들어오셈

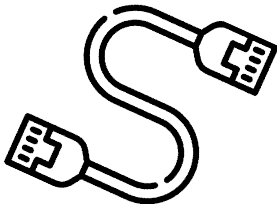
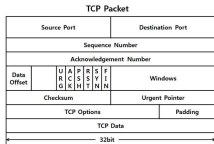


남



4. 이더넷 카드는 TCP/IP 패킷을 바로 윗 계층(Internet Layer)으로 보내는데, IP 패킷을 분석해서 이 패킷이 어디에서 왔으며, 도착지가 어디인지를 판단한다.

TCP/IP 패킷

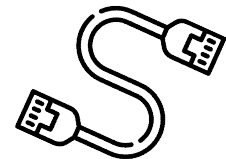


우리 보내줘요

너네 어디서 왔어

Internet Layer : 애들아
얼렁 와,,,

6. 목적지가 자신이면 이것을 다시 Transport Layer로 보내고, TCP 프로토콜을 사용하여 메시지를 검사, 다시 Application Layer에게 보낸다.



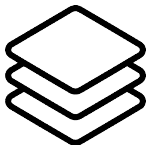
여기 맞긴 한데 입국
심사 받아야댐 ㄱㄷ

TCP/IP 패킷

TCP Packet									
Source Port					Destination Port				
Sequence Number									
Acknowledgement Number									
Data Offset	U	A	P	R	S	F	Windows		
	R	C	S	S	V	I			
	C	H	T	N	S				
Checksum					Urgent Pointer				
TCP Options					Padding				
TCP Data									
Size									



엉엉 할 일이 너무 마나



나 Transport

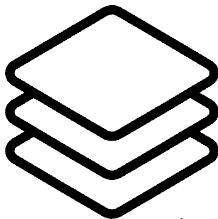


나 Application

7. Application Layer에서는 웹서버가 통신 메시지를 HTTP 프로토콜에 준하여 검사한다.

TCP/IP 패킷

Source Port					Destination Port				
Sequence Number									
Acknowledgement Number									
Data Offset		U	A	P	R	S	F	Windows	
		R	C	S	S	V	I		
		G	K	H	T	N	N		
Checksum					Urgent Pointer				
TCP Options							Padding		
TCP Data									
32bit									



헛

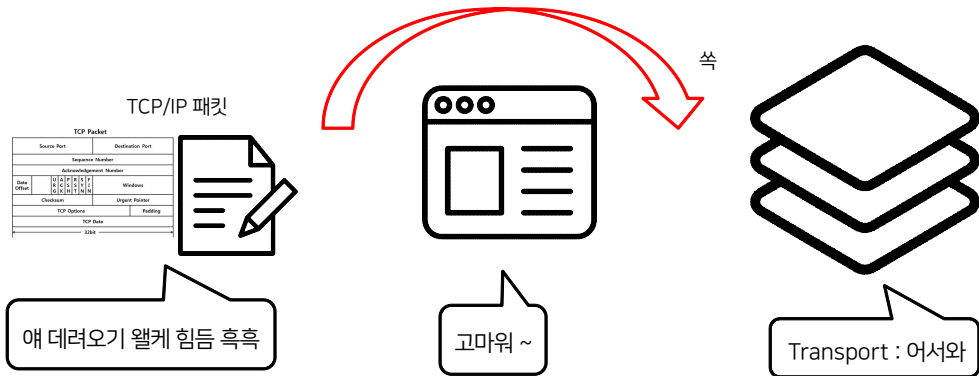
http://



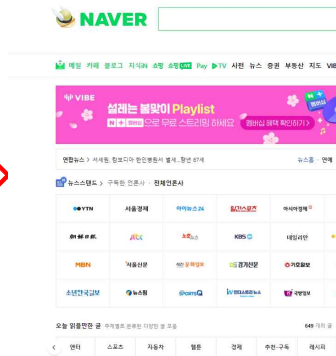
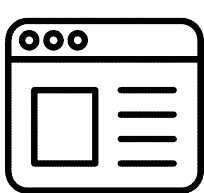
진자 언제 도착 하나 우리

Application : 우리 HTTP 까다로우니까
검사 좀 할게여

8. 사용자가 요청한 웹페이지를 읽어들이고 후 Transport 계층으로 보낸다.



9. 웹 브라우저는 웹페이지를 받아서 렌더링 작업을 거친 후 화면에 뿌린다.



뵙

뵙

TCP는 연결지향 프로토콜이다

- 데이터가 전송되기 전에 브라우저와 서버 간의 연결을 성립하는 과정이 이루어지게 된다.
- 호스트 간에 연결 설정 방법을 **3way handshake** 라고 부르며 이는 SYN/ACK 패킷을 통해 이루어진다.
- SYN 패킷 ? 동기화(SYNchronize)를 의미
- ACK 패킷은 확인(ACKnowledgement)을 의미

1번
어이 서버, 잘 있나?

2번
당연하지 브. 데이터 주셈

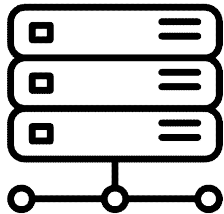
3번
ㅇㅋ 보냄



통신선로 개설



브라우저

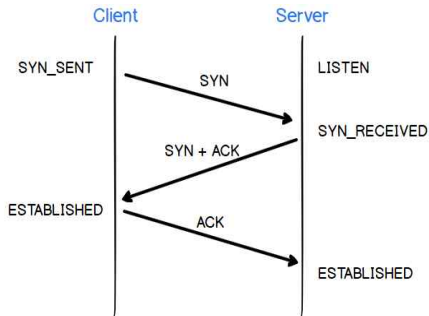


서버

통신선로 개설을 위해 3번의 데이터 전송이 일어남

이것이 바로 **3way handshake**

- 연결이 이루어지면 모든 정식 데이터는 연결된 통신선로를 통해서 교환되게 되며, 이러한 이유로 **연결 지향 프로토콜**이라고 부른다.
- UDP는 이러한 과정이 없이 단순히 데이터만을 전송하기 때문에 “데이터 그램” 프로토콜이라고 부른다.



1. LISTEN : 서버가 클라이언트의 연결요청을 기다리고 있다.

2. SYN_SENT : 클라이언트가 능동적으로 서버에게 연결요청을 하자고 시퀀스 번호를 생성하여 SYN 패킷에 담아 보낸다. (능동 개방)

3. SYN_RECEIVED : SYN 패킷을 받은 서버는 자신만의 시퀀스 번호를 생성하여 SYN 패킷에 담고 클라이언트의 SYN 패킷에 있는 시퀀스 번호에 1을 더해서 ACK 패킷에 담아 같이 보낸다.

4. 클라이언트 ESTABLISHED : SYN+ACK 패킷을 받은 클라이언트는 ACK 패킷의 시퀀스 번호를 보고 자신이 보낸 시퀀스 번호와 차이가 1임을 확인한다. 차이가 1이라면 제대로 연결되었다고 판단하고 서버의 SYN 패킷에 있는 시퀀스 번호에 1을 더해 ACK 패킷에 담아 보낸다.

5. 서버 ESTABLISHED : 클라이언트의 ACK 패킷을 받고 그 안의 시퀀스 번호가 보냈던 SYN 패킷의 시퀀스 번호 + 1이라면 연결이 되었다고 판단한다. 이후부터 본격적인 통신을 할 수 있게 된다.

발표자가 모르기 때문에 질문을 못 받습니다.

근데 질문 남겨주시면 열심히 찾아서 답변 업로드 하겠음 (하트)

감삼당