

● OSI 7 계층(OSI 7 Layer): 네트워크 프로토콜이 통신하는 구조를 7개의 계층으로 분리하여 각 계층간 상호 작동하는 방식을 정해 놓은 것 이다. 이는 ISO(국제표준화기구)에서 개발한 모델이다. 7단계 중 특정한 곳에 이상이 생기면 다른 단계의 장비 및 소프트웨어를 건드리지 않고도 이상이 생긴 단계만 고칠 수 있다. Physical Layer, DataLink Layer, NetWork Layer, Transport Layer, Session Layer, Presentation Layer, Application Layer.

● TCP/IP 5계층: 수많은 프로그램들이 인터넷으로 통신하는데 기반이 되는 프로토콜이다. 현대의 인터넷은 이 모델을 따르고 있다. Physical Layer, DataLink Layer, NetWork Layer, Transport Layer, Application Layer.

▶ 1계층. Physical Layer: 물리적으로 연결된 두 대의 컴퓨터가 0과 1의 나열을 주고받을 수 있게 해주는 모듈

원본 data > 1계층 encoder > 아날로그 신호
아날로그 신호 > 1계층 decoder > 원본 data

→ 구현: PHY칩. 하드웨어로 구현되어 있다.

전선으로 연결되어 있는 컴퓨터들. 공통된 전선을 박스에 둔 게 더미 박스. 데이터를 전송할 때 해당 컴퓨터로만 전송하려면 스위치를 사용. 스위치와 스위치를 연결해서 서로 다른 네트워크에 속한 컴퓨터끼리 통신이 가능하게 해주는 장비를 라우터라고 한다. 즉, 라우터는 스위치 역할도 하고 라우터 역할도 한다. 이때, 이 스위치는 공유기에 해당된다. 공유기 — 라우터 — 공유기. 이런 식으로 전 세계의 컴퓨터들이 계층구조로 연결되어 있는 것을 인터넷이라고 한다.

▶ 2계층. Data-link Layer: 같은 네트워크에 있는 여러 대의 컴퓨터들이 데이터를 주고받기 위해서 필요한 모듈. Framing(송신자가 데이터 앞뒤에 특정한 비트열을 붙이는 것)은 Data link layer에 속하는 작업들 중 하나.

Data > 2계층 encoder > framing+data
framing+data > 2계층 decoder > data

→ 구현: 랜카드에 구현되어 있다. 하드웨어적으로 구현.

▶ 3계층. Network Layer: 수많은 네트워크들의 연결로 이루어지는 inter-network 속에서 어딘가에 있는 목적지 컴퓨터로 데이터를 전송하기 위해 IP 주소를 이용해서 길을 찾고(routing), 자신의 다음 라우터에게 데이터를 넘겨주는 것(forwarding)

발신 컴퓨터 data를 보낼 때 그 앞에 수신 컴퓨터의 고유한 주소, 즉, IP주소를 붙여서 보낸다. 수신 컴퓨터의 IP를 알고 있어야 데이터를 전송할 수 있다. 데이터에 IP 주소를 붙인 것을 패킷이라고 부른다. 컴퓨터에서 패킷 전송 - 라우터가, 목적지 IP주소 확인 - 데이터 재포장 후 라우터 마에 전송 - 라우터 마, 목적지 주소 확인 후 어느 전선으로 패킷을 내보내야 하는 알아내야 함 . 라우터 바에게 넘겨주어야 한다고 판단 후 다시 포장해서 라우터 라에 보냄. - 컴퓨터 B에 전송.

data > 3계층 encoder > IP주소+data (객체)
IP주소+data (객체) > 3계층 decoder > data

→ 구현: 운영체제 커널에 소프트웨어적으로 구현되어 있다.

▶ **4계층. Transport Layer:** Port번호를 사용하여 도착지 컴퓨터의 최종 도착지인 프로세스까지 데이터가 도달하게 하는 모듈

데이터를 받고자 하는 프로세스들은 포트 번호 (port number)라는 것을 가져야 한다. 이는 하나의 컴퓨터에서 동시에 실행되고 있는 프로세스들 서로 겹치지 않게 가져야 하는 정수 값이다. 한편, 송신자는 데이터를 보낼 때 데이터를 받을 수신자 컴퓨터에 있는 프로세스의 포트 번호를 붙여서 보낸다.

data > 4계층 encoder > Port+data

Port+data > 4계층 decoder > data

→ 구현: 운영체제의 커널에 소프트웨어적으로 구현되어 있다.

▶ **5계층. Application Layer:** 최상위 계층으로 사용자가 네트워크 자원에 접근하는 방법을 제공한다. 최종적으로 사용자가 볼 수 있는 유일한 계층으로 모든 네트워크 활동의 기반이 되는 인터페이스를 제공하는데, 즉 사용자가 실행하는 응용 프로그램들이 계층 7에 속한다고 보면 된다. 예를 들면 가상 터미널인 텔넷(telnet), 구글의 크롬(chrome), 이메일(전자우편), 데이터베이스 관리 등의 서비스를 제공한다. 사용자와 가장 가까운 계층이다.

최종 목적지로서 HTTP, FTP, SMTP, POP3, IMAP, Telnet 등과 같은 프로토콜이 있다. 해당 통신 패킷들은 방금 나열한 프로토콜에 의해 모두 처리되며 우리가 사용하는 브라우저나, 메일 프로그램은 프로토콜을 보다 쉽게 사용하게 해주는 응용프로그램이다. 한마디로 모든 통신의 양 끝단은 HTTP와 같은 프로토콜이지 응용프로그램이 아니다. 응용 계층(Application layer)은 응용 프로세스와 직접 관계하여 일반적인 응용 서비스를 수행한다. 일반적인 응용 서비스는 관련된 응용 프로세스들 사이의 전환을 제공한다.

TCP/IP 소켓 프로그래밍: 운영체제의 Transport Layer에서 제공하는 API를 통해서 통신 가능한 프로그램을 만드는 것을 TCP/IP 소켓 프로그래밍 또는 네트워크 프로그래밍이라고 한다. 소켓 프로그래밍 만으로도 클라이언트, 서버 프로그램을 따로 따로 만들어서 동작시킬 수 있고, TCP/IP 소켓 프로그래밍을 통해서 누구나 자신만의 Application Layer 인코더와 디코더를 만들 수 있다. 누구든 자신만의 Application Layer 프로토콜을 만들어서 사용할 수 있다는 뜻이다.

대표적인 Application Layer 프로토콜인 HTTP

서버 컴퓨터 데이터 전송

위해 data > HTTP encoder > Status code + 위해

Status code + 위해 > HTTP encoder > 위해

소프트웨어 아키텍처 중에 Layer Architecture을 따르는 대표적인 예가 네트워크 시스템이다.

Encoding

· 5계층 data > HTTP encoder > Status code + data

· 4계층 Status code + data > 4계층 encoder > Port + Status code + data

송신자는 데이터를 보낼 때 데이터를 받을 수신자 컴퓨터에 있는 프로세스의 포트 번호를 붙여서 보낸다.

· 3계층 Port + Status code + data > 3계층 encoder > IP주소+ Port + Status code + data

데이터 앞에 수신 컴퓨터의 고유한 주소, 즉, IP주소를 붙여서 보낸다.

· 2계층 IP주소+ Port + Status code + data > 2계층 encoder > framing + IP주소+ Port + Status code + data + framing

데이터 앞뒤에 특정한 비트열을 붙인다

· 1계층 framing + IP주소+ Port + Status code + data + framing > 1계층 encoder > 아날로그 신호

데이터를 아날로그 신호로 바꾼다

수신 컴퓨터는 붙인 정보들(Header)를 역순으로 하나씩 제거하며 떼내는 Decoding 작업을 통해 상위 계층으로 데이터를 전달하고, 최종적으로 원본 데이터를 수신한다.

OSI 7계층

- ▶ 5계층. Session Layer: 세션을 열고 닫고를 제공하는 메커니즘의 계층 세션 복구도 지원한다(체크 포인트를 이용해 동기화한다)
- ▶ 6계층. Presentation layer: 데이터의 변환 압축, 및 데이터의 암호화
- ▶ 7계층. Application Layer: 프로토콜들이 속한 계층

Resources: <https://youtu.be/1pfTxp25MA8>, http://wiki.hash.kr/index.php/OSI_7_%EA%B3%84%EC%B8%B5, <https://shlee0882.tistory.com/110>
