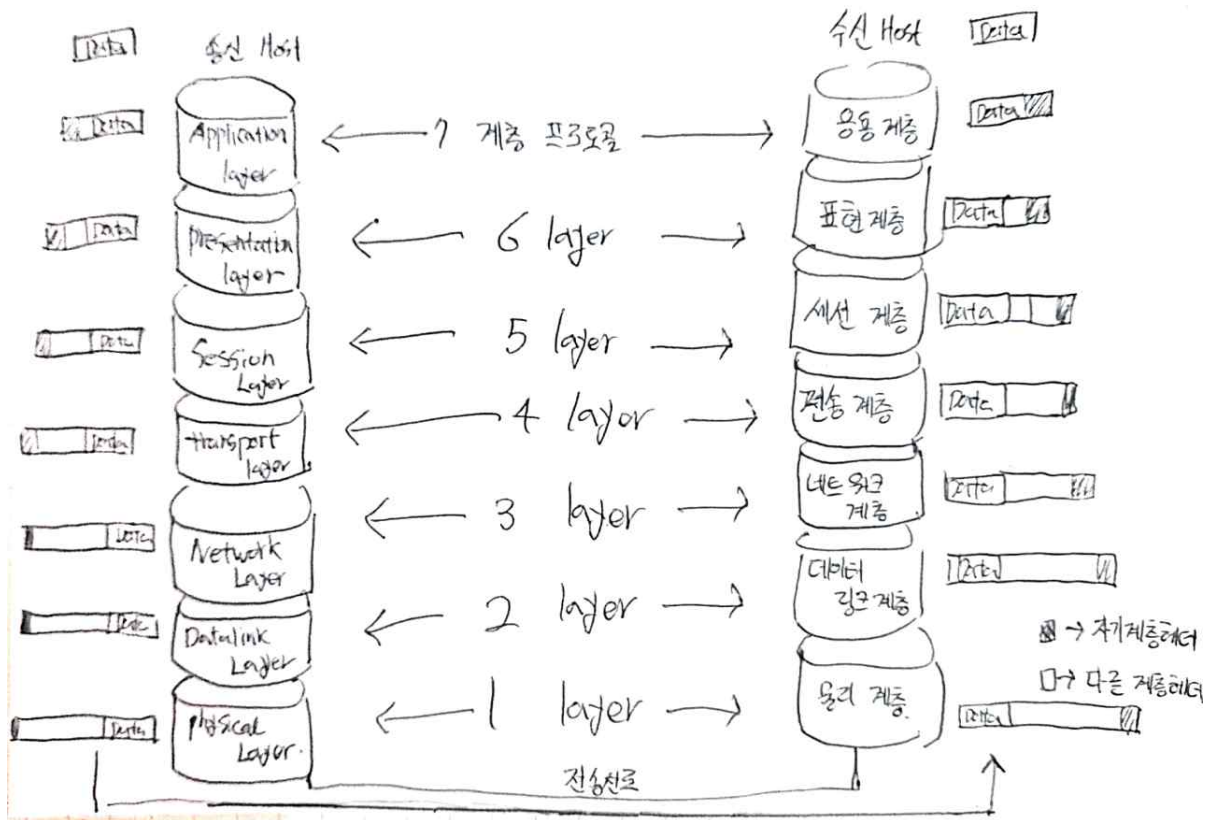


Computer Science 정리

1. 컴퓨터 네트워크

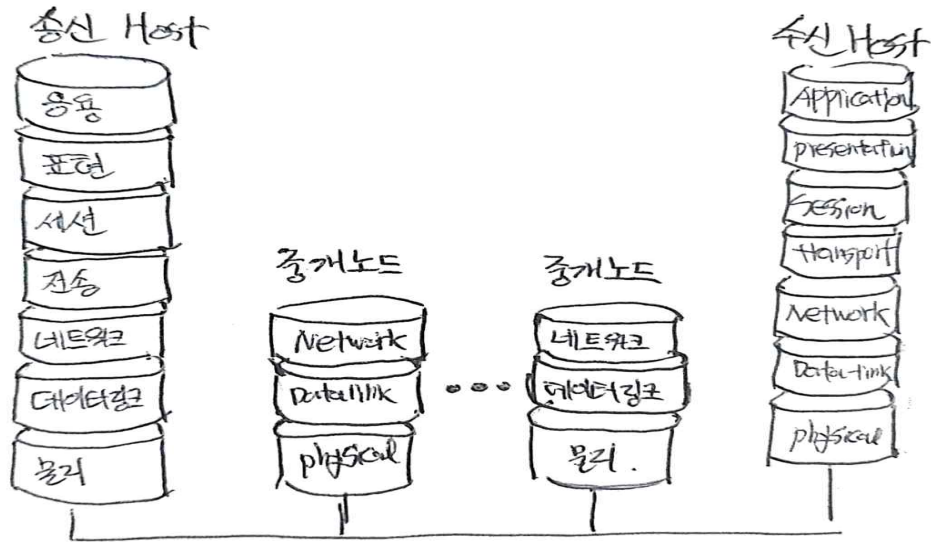
1. OSI 참조 모델

- 통신 기술의 도입과 통신기능의 확장을 쉽게 하려고 프로토콜을 몇 개의 계층으로 나누는 것을 '계층화'라고 한다.
- 통신 기능을 7계층으로 분류하여 각 계층마다 프로토콜을 규정한 규격을 OSI 모델이라 한다
- OSI 7 layer 동작



- * 송신 host에서 수신 host로 계층을 이동할때마다 각 계층의 header가 data에 추가(송신)되고 처리(수신)된다.
- * 결과적으로 Data는 Application(송신) -> Application(수신) 으로 이동한다.

- OSI 7 layer Model의 중계기능



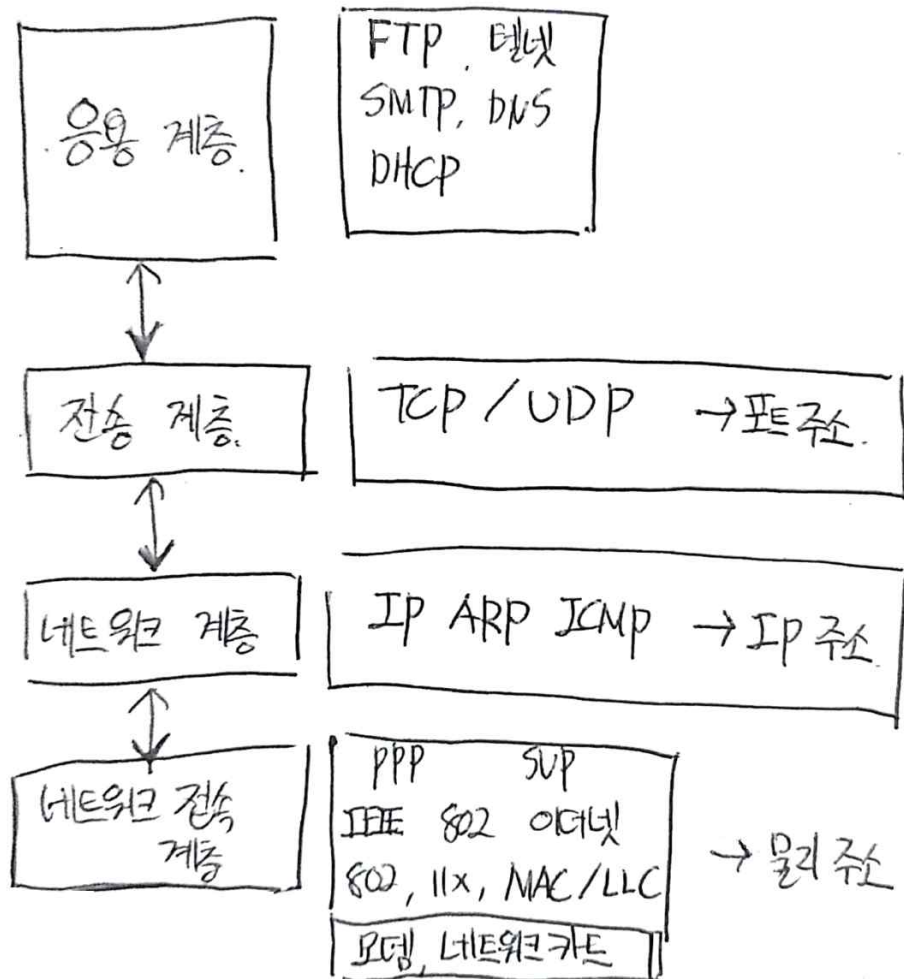
* 송/수신 시에만 Application 까지 데이터가 이동 , 데이터 전송 중에는 네트워크(3layer) 까지만 이동한다.

- OSI 7layer Model의 계층별기능

물리 계층(physical)	<ul style="list-style-type: none"> - 전송 매체의 물리적 인터페이스에 관한 사항을 기술함. - 데이터 전송 속도, 송수신 호스트 사이의 클럭 동기화 방법, 물리적 연결 형태 등 기술
데이터 링크 계층 (Data link)	<ul style="list-style-type: none"> - 데이터의 물리적 전송 오류를 해결 - 프레임(frame) : 전송하는 데이터의 명칭
네트워크 계층 (Network)	<ul style="list-style-type: none"> - 송신 호스트가 전송한 데이터가 어떤 경로를 통해 수신 호스트에 전달 되는지 결정하는 라우팅 문제를 처리 - 호스트 구분을 위한 주소 개념 필요 (ex : ip주소) - 패킷 : 전송 데이터의 명칭 - 혼잡 제어 : 데이터 전송 경로 선택에 따라 네트워크 혼잡에 영향
전송 계층 (Transport)	<ul style="list-style-type: none"> - 송신 프로세스와 수신 프로세스를 직접 연결하는 단대단(End-to-End) 통신 기능 제공
세션 계층 (Session)	<ul style="list-style-type: none"> - 송수신 호스트 사이의 대화 제어를 비롯한 상호 배타적인 동작을 제어하기 위한 큰 제어, 일시적인 전송 장애를 해결하기 위한 동기 Synchronization 기능 등 제공
표현 계층 (Presentation)	<ul style="list-style-type: none"> - 데이터의 의미와 표현 방법을 처리, 데이터를 코딩(Coding)하는 문제를 다룸.
응용 계층 (Application)	<ul style="list-style-type: none"> - 최상위, 다양하게 존재하는 응용 환경에서 공통으로 필요한 기능을 다룸 - 대표적인 인터넷 서비스, FTP, Telnet, 전자메일

2. TCP/IP

- 네트워크는 서로 다른 기종의 컴퓨터로 구성되어 있어 각 네트워크 간에 공통으로 사용할 수 있는 프로토콜이 필요.
- 이에 따라 인터넷에서 컴퓨터간의 통신이 가능하도록 표준화 하여 채택한 규약이 TCP/IP
- 네트워크와 네트워크를 연결하는 데 사용하는 프로토콜인 TCP/IP는 '전송 제어 프로토콜' 과 '인터넷 프로토콜'을 의미한다. (Transmission Control Protocol / Internet Protocol)
- 인터넷에서 사용하는 응용 프로그램은 대부분 이 TCP/IP 프로토콜을 이용하여 데이터를 교환한다.



물리 주소	- 물리(MAC) 주소는 링크 주소 또는 통신망에서 정의된 노드의 주소, 이더넷 , 네트워크 인터페이스카드(NIC) , 6바이트(48비트) 주소 등을 말한다.
인터넷 주소	- 인터넷에서는 기존 물리 주소와는 별도로 각 호스트를 식별할 수 있는 유일한 주소를 지정해야 한다.
포트 주소	- 수신지 컴퓨터까지 전송하려면 IP주소와 물리 주소가 필요하다. - 인터넷 통신의 최종 목적은 한 프로세스가 다른 프로세스와 통신 할 수 있도록 하는 것이다.

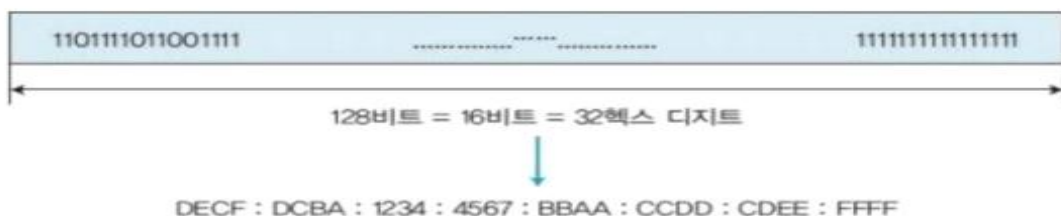
- TCP/IP vs OSI 7 Layer

유사점	차이점
1. 계층 모델을 사용한다. 2. 응용 프로그램 계층(Application) 3. 트랜스포트/네트워크 계층과 호환되는 계층 사용 4. 패킷 스위칭 기술 기반.	1. TCP/IP 모델의 응용 프로그램 계층은 OSI 계층의 표현 계층과 세션 계층을 포함. 2. TCP/IP 모델의 네트워크 액세스 계층은 OSI모델의 데이터 링크 계층과 물리 계층을 포함한다. 3. TCP/IP 프로토콜은 인터넷 발전에 표준..?

- * 패킷 스위칭 : 송신 측에서 데이터를 일정한 크기의 패킷으로 분해해서 전송하고 , 수신측에서 이를 조립
- 라우터를 사용해 최적의 경로를 설정 데이터 전송 , 우선순위를 표기하여 중요 패킷 식별 가능.
 - 순서 제어 , 트래픽 제어, 에러 제어
 - 고신뢰성 , 고품질, 고효율, 이기종간 단말 통신 기능 , 실시간 적합 (vs 서킷 스위칭(속도와 성능이 일정))

- IPv6

IPv4의 한계	<ul style="list-style-type: none"> - IPv4 주소는 32비트로 구성되며, 네트워크 ID와 호스트 ID로 나뉜다. - 또한 2-level 주소 구조(Net ID, Host ID)로 되어 있다. - IPv4 주소는 산술적으로 주소를 43억개 할당 할 수 있지만, 클래스 별 주소 분류 방식 때문에 사용하지 않는 주소가 많았음. - IoT 환경에서는 각각의 단말기마다 IP가 부여되는 환경이 필요하므로 기존 IPv4를 사용하면 IP주소가 부족하게 됨
IPv6	<ul style="list-style-type: none"> - 주소 체계가 128비트로 구성되며, 긴 주소를 쉽게 읽을 수 있도록 16키트씩 :(콜론) 으로 나누어 각 필드를 16진수로 표현하는 방법을 사용 - 기존 IPv4 주소도 IPv6 주소로 표현 할 수 있음. 하위 32비트에는 IPv4 주소를 그대로 채우고 상위 비트는 모두 0으로 채우는 방식을 사용
IPv6 주소 체계	<ul style="list-style-type: none"> - IPv6 주소는 16바이트(128비트)로 구성되며, 주소를 읽기 쉽도록 16진수 콜론으로 표현 - 128비트는 길이가 2바이트인 영역 8개로 나뉘며 , 16진수 표기법에서 2바이트는 16진수 4개로 표현하여 전체적으로 16진수 32개로 표현한다.



- IPv4와 IPv6의 패킷 헤더와 비교

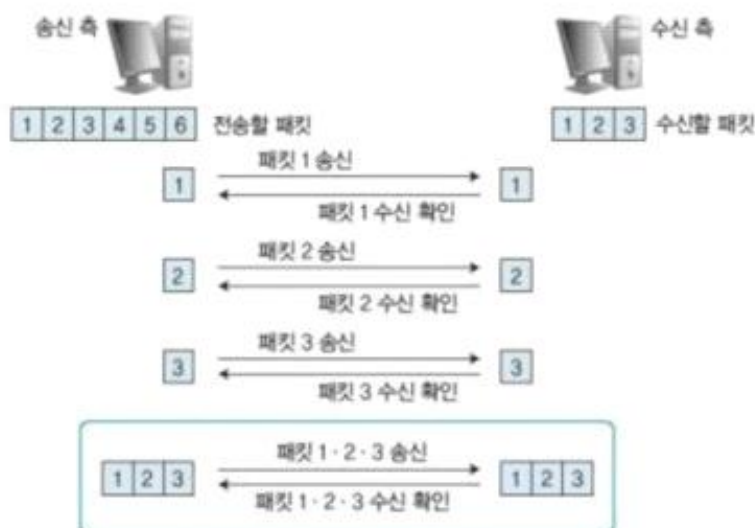
- IPv6은 헤더의 길이가 고정되어 있어 헤더 길이 필드가 제거되고 서비스 유형 필드도 제거
- 그 기능을 우선순위와 흐름 레이블 필드가 대체
- 총 길이 필드 또한 제거되어 페이로드 길이 필드로 대체
- 식별,플래그,옵션 필드는 기본 헤더에서 제거되고, 이 필드는 확장 헤더에 포함

항목	IPv4	IPv6
주소 크기	32비트	128비트
플래그 레이블	X	O
헤더 검사합	O	X
단편	모든 데이터그램	옵션

- TCP 흐름 제어

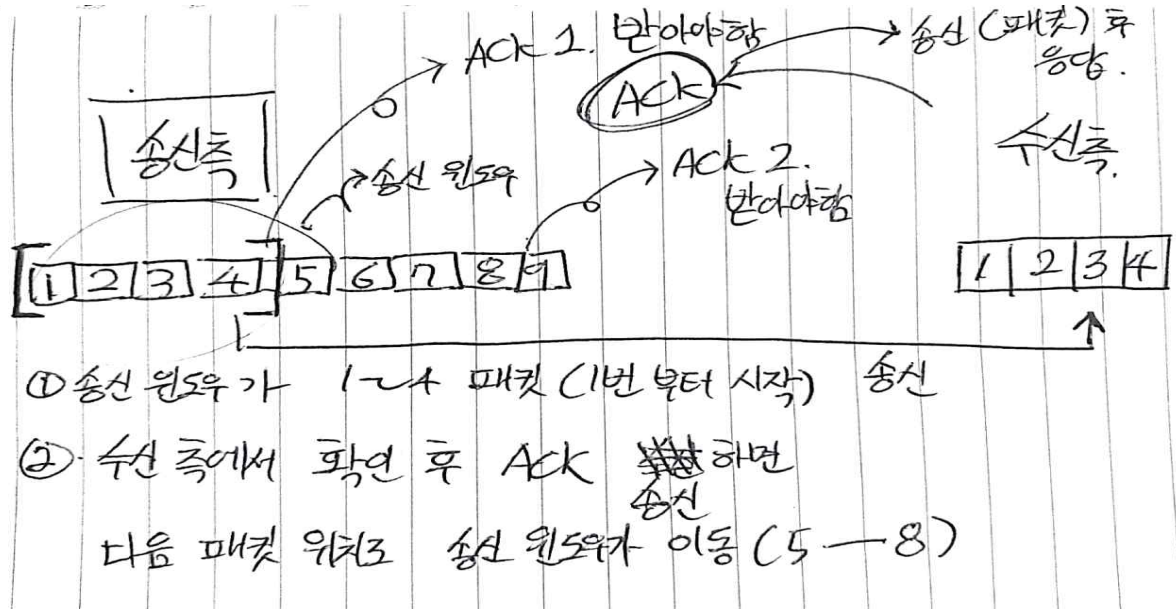
-TCP 프로토콜의 특징

- TCP는 원래의 데이터를 패킷이라는 단위로 분할하여 전송
- 수신측의 컴퓨터는 수신한 패킷을 재결합하여 다시 원래의 데이터로 만들어야함
- 패킷을 전송할 때 네트워크 상황에 따라 패킷의 도착 순서가 바뀔 수 있고, 중간에서 사라지는 패킷을 한 번에 하나씩 송수신하는 것보다 여러 패킷을 한 번에 송수신 한 후 수신 확인 신호를 전송 하는 것이 훨씬 효율적임. (중요)
- 한 번에 전송하는 패킷의 개수를 ‘윈도우 크기’ 또는 ‘수신 버퍼의 크기’ 라고 한다.
- 즉, 데이터를 전송할 때 한번에 전송할 수 있는 전체 패킷의 크기를 ‘TCP 윈도우 크기’ 라고 하며 윈도우 크기가 크면 한 번에 여러 패킷을 전송할 수 있다.



- 슬라이딩 윈도우(Sliding Window)

- 송신측 컴퓨터는 윈도우 크기에 따라 1~4번까지 패킷을 전송하고, 수신 측 컴퓨터에서 수신 확인 ACK신호를 수신하면 ACK 신호에서 요청한 5번 패킷 위치로 송신 윈도우를 오른쪽으로 이동
- 다시 윈도우 안의 패킷들을 전송하고, 수신 측 컴퓨터에서 ACK신호를 수신하면 ACK 신호에서 요청한 9번 패킷 위치로 송신 윈도우를 옮긴다.
- 이처럼 송신 버퍼 역할을 하려고 송신 윈도우를 이동하는 방법을 '슬라이딩 윈도우' 라고 한다.



- * 송신 윈도우에 담긴 패킷 전송 -> 수신확인 후 수신측에서 ACK 전송 -> ACK 확인 후 송신 윈도우 이동
- * ACK가 장시간 수신되지 않거나 NAK가 수신 될 경우 전송했던 데이터를 재전송 한다.

- 웹의 동작

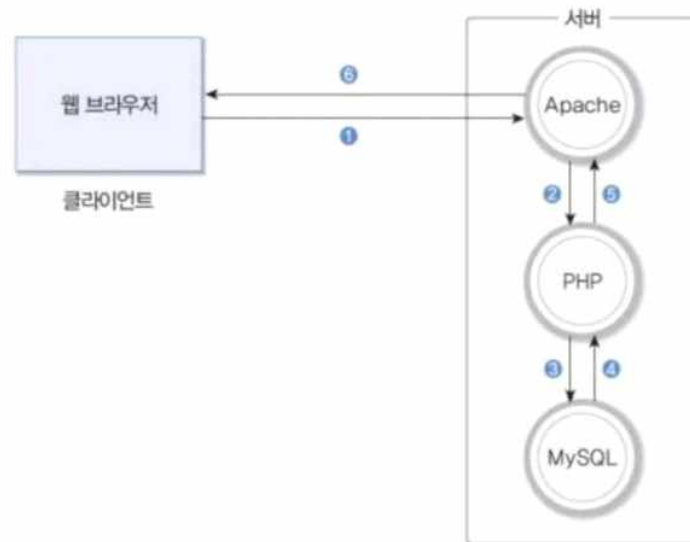
- 웹(WWW , World Wide Web) 서버의 TCP포트 번호 : 80번으로 지정

- 웹 서비스의 동작 원리

1. 웹 브라우저에 URL 주소 입력
2. 클라이언트는 서버 호스트 이름을 DNS 서버에 전송, 웹 주소의 IP를 얻음
3. IP 주소와 Well - Known 포트 80번을 사용하여 웹 서버와 TCP 연결을 시도
4. TCP 연결이 설정되고 클라이언트가 서버에 GET 명령을 전송
5. 서버가 요청한 웹 문서를 웹 브라우저에 회신
6. 둘 사이의 TCP 연결을 해제
7. 웹 브라우저는 해당 파일의 내용을 사용자가 볼 수 있게 화면에 표시

- 웹 서비스의 동작 원리 (예시 : Apache + Php + Mysql) ,APM의 동작 원리

1. 웹 브라우저가 Apache에 웹 문서 요청
2. Php 코드 처리 필요 시 PHP에 요청
3. 데이터베이스 처리 필요시 Mysql에 요청
4. 데이터베이스 결과 회신
5. PHP가 실행 결과인 HTML 코드 회신
6. 웹 문서를 웹 브라우저에 회신



- * 웹 서버(Apache)를 이용해 처리하는 예시
- * 추가적으로 WebContainer기능을 함께 사용하는 WAS(Web Application Server)를 이용하면 더 좋음
- * WebServer에서 정적 처리를 하고 Web Container에서 동적 처리(DB 연동 , 계산 등)을 하면 효율이 높아짐
- * 대표적인 예시로 Tomcat이 있음

