Network

프로토콜

- 개요
  - ✓ 규정이나 규약과 관련된 내용
  - ✓ 네트워크에서 통신할 때의 규약을 프로토콜이라는 용어를 사용
  - ✓ 프로토콜은 어떤 표준 협회나 워킹 그룹이 만들었는지 또는 어떤 회사에서 사용하느냐에 따라 특징이 많이 달라지고 다양한 프로토콜이 존재
  - ✓ 최근에는 복잡하고 산재되어 있던 여러 가지 프로토콜 기술이 이더넷-TCP/IP 기반 프로토콜들로 변경되고 있음
    - 물리적 측면: 데이터 전송 매체, 신호 규약, 회선 규격 등으로 이더넷이 널리 쓰임
    - 논리적 측면: 장치들 끼리 통신하기 위한 프로토콜 규격으로 TCP/IP가 널리 쓰임
  - ✓ 적은 컴퓨팅 자원과 매우 느린 네트워크 속도를 이용해 최대한 효율적으로 통신하는 것이 목표이다 보니 대부분의 프로토콜이 문자 기반이 아닌 2진수 비트(bit) 기반으로 만들어졌음
  - ✓ 최소한의 비트로 내용을 전송하기 위해서는 매우 치밀하게 서로 간의 약속을 정의해야 함
  - ✓ 몇 번째 전기 신호는 보내는 사람 주소 몇 번째 전기 신호는 받는 사람 주소 몇 번째 전기 신호는 상위 프로토콜 지시자 등과 같이 미리 까다롭게 약속하고 그 약속을 철저히 지켜야 통신을 수행할 수 있었는데 애플리케이션 레벨의 프로토콜은 비트 기반이 아닌 문자 기반 프로토콜들이 많이 사용되고 있는데 HTTP와 SMTP와 같은 프로토콜이 대표적
  - ✓ 비트로 메시지를 전달하지 않고 문자 자체를 이용해 헤더와 헤더 값 데이터를 표현하고 전송

프로토콜

❖ 개요

✓ HTTP 프로토콜의 헤더

GET /api HTTP/1.1

Accept: text/html, application/xhtml+xml, */*

Referer: http://zigispace.net/

Accept-Language: ko-KR

User-Agent: Mozilla/5.0 (Windows NT 6.1 ; W0W64 ; Trident/7.0 ; TCO_20181006113830 ;

rv : 11.0) like Gecko

Accept-Encoding: gzip, deflate

Host: theplmingspace.tistory.com

DNT: 1

Connection: Keep-Alive

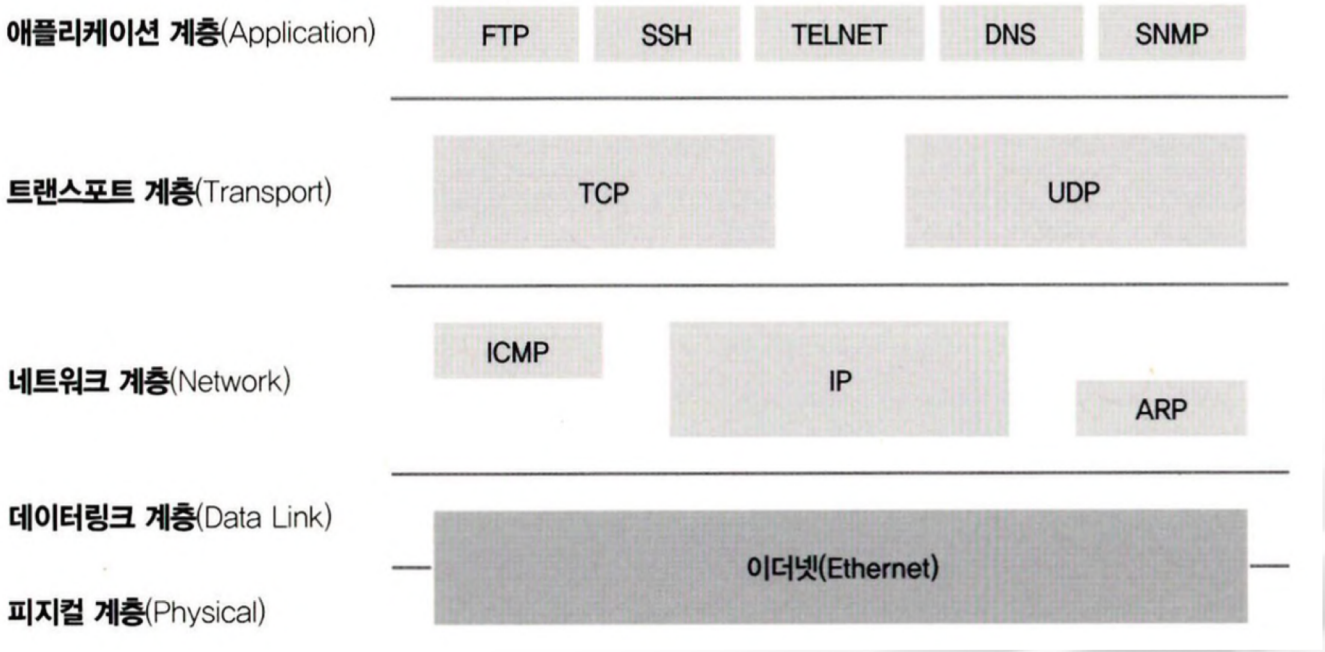
프로토콜

- ❖ 개요
  - ✓ HTTP 프로토콜의 헤더
    - 텍스트(Text) 파일과 같은 데이터가 전달되기 때문에 효율성은 비트 기반 프로토콜보다 떨어지지만 다양한 확장이 가능
    - 일반적으로 TCP/IP는 프로토콜이라고 부르지 않고 프로토콜 스택이라고 부르는데 TCP와 IP는 별도 계층에서 동작하는 프로토콜이지만 함께 사용하고 있는데 이런 프로토콜 묶음을 프로토콜 스택이라고 부름
    - TCP/IP 프로토콜 스택에는 TCP 와 IP 뿐 아니라 UDP, ICMP, ARP, HTTP, SMTP, FTP와 같은 매우 다양한 애플리케이션 레이어 프로토콜들이 있음
    - TCP/IP 프로토콜 스택은 총 4개 부분으로 나뉨
    - 물리 부분인 이더넷 외에 데이터가 목적지를 찾아가도록 해주는 네트워크 계층 및 잘린 패킷을 데이터 형태로 잘 조합하도록 도와주는 전송 계층과 애플리케이션 계층으로 구성

프로토콜

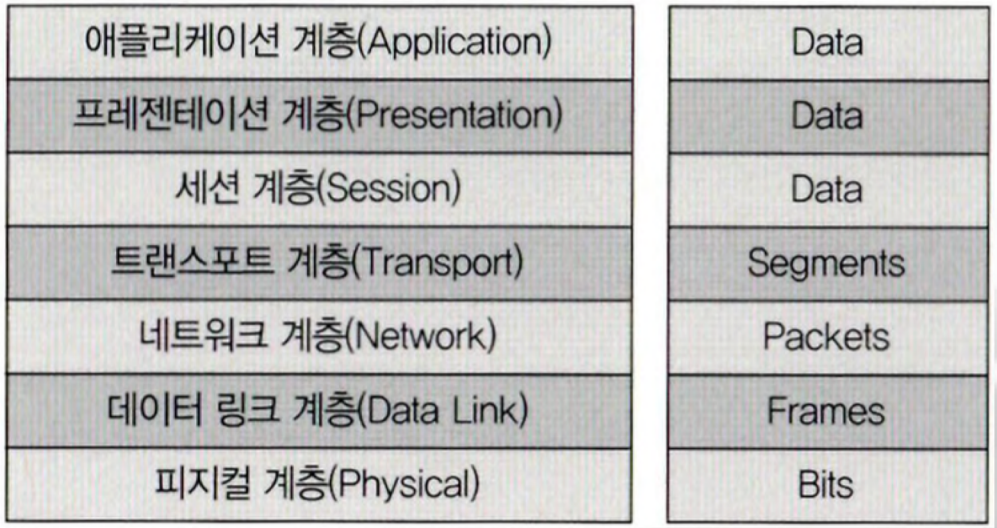
❖ 개요

✓ TCP/IP 프로토콜 스택



OSI 7계층 과 TCP/IP

- OSI 7계층
  - ✓ 과거에는 통신용 규약이 표준화되지 않았고 각 벤더에서 별도로 개발했기 때문에 호환되지 않는 시스템이나 애플리케이션이 많았고 통신이 불가능했는데 이를 하나의 규약으로 통합하려는 노력이 현재의 OSI 7계층으로 남아 있음
  - ✓ OSI 7계층이 네트워크 동작을 나누어 이해하고 개발하는데 많은 도움이 되므로 네트워크의 주요 레퍼런스 모델로 활용되고 있지만 현재는 대부분의 프로토콜이 TCP/IP 프로토콜 스택 기반으로 되어 있음

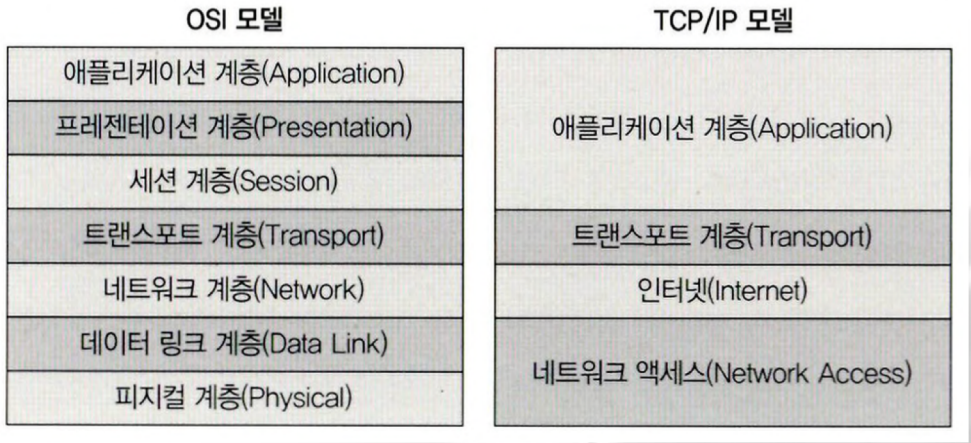


OSI 7계층 과 TCP/IP

- OSI 7계층
  - ✓ 복잡한 데이터 전송 과정을 OSI 7계층으로 나누어 보면 이해하기 쉬움
  - ✓ 계층별로 표준화된 프로토콜 템플릿을 통해 네트워크 프로토콜을 전부 개발하는 대신 계층별로 프로토콜을 개발해 네트워크 구성 요소들을 모듈화 할 수 있음
  - ✓ 모듈화된 요소는 기존에 개발된 프로토콜과 연동해 사용할 수 있음
  - ✓ OSI 7계층은 다시 두 가지 계층으로 나눌 수 있음
    - 1~4 계층: Data Flow Layer /하위 계층(Lower Layer)
    - 5~7 계층: Application Layer /상위 계층(Upper Layer)
  - ✓ Data Flow 계층은 데이터를 상대방에게 잘 전달하는 역할을 수행하는데 애플리케이션 개발자는 Application Layer 프로토콜을 개발할 때 하위 Data Flow 계층을 고려하지 않고 데이터를 표현하는 데 초점을 맞춤
  - ✓ 애플리케이션 계층은 애플리케이션 개발자들이 고려해야 할 영역이므로 네트워크 엔지니어는 이 부분에 대해서는 일반적으로 심각하게 고려하지 않음
  - ✓ 이런 이유로 애플리케이션 개발자는 하향식(Top-Down) 형식으로 네트워크를 바라보고 네트워크 엔지니어는 상향식(Bottom-Up) 형식으로 네트워크를 인식

OSI 7계층 과 TCP/IP

- TCP/IP 프로토콜 스택
  - ✓ 현대 네트워크는 대부분 TCP/IP와 이더넷으로 이루어져 있음
  - ✓ 일부 특수한 환경에서는 다른 프로토콜이 사용되기도 하지만 다양한 기술과 프로토콜 중 어느 것을 선택해야 할지 고민하던 과거와는 상황이 매우 다름
  - ✓ TCP/IP와 이더넷이 개발된 것은 매우 오래전인데 몇 번의 큰 기술 발전을 거쳐 현재는 값싸고 성능이 우수한 TCP/IP와 이더넷이 되었음
  - ✓ 기술과 표준을 만들 때 만들어진 역사적 배경이나 만든 조직 또는 프로토콜이 만들어진 목표에 따라 성향이 많이 반영되는데 TCP/IP는 이론보다 실용성에 중점을 둔 프로토콜



OSI 7계층 과 TCP/IP

- TCP/IP 프로토콜 스택
  - ✓ OSI 레퍼런스 모델은 7계층으로 이루어진 반면 TCP/IP 모델은 4계층으로 구분
  - ✓ 애플리케이션 개발자가 고려해야 할 부분과 서버 엔지니어나 네트워크 엔지니어가 고려해야 할 부분이 구분되었는데 TCP/IP 모델은 그 구분이 더 확연히 드러남
  - ✓ 상위 3개 계층을 하나의 애플리케이션 계층으로 묶고 1, 2 계층 물리 계층 과 데이터 링크 계층을 하나의 네트워크 계층으로 구분
  - ✓ 현실에 쉽게 반영하도록 간단히 구분하는 TCP/IP 프로토콜 스택의 성향이 이곳에서도 드러남

OSI 7계층 별 이해

- 1 계층(물리 계층)
  - ✓ 1 계층은 물리 계층으로 물리적 연결과 관련된 정보를 정의
  - ✓ 전기 신호를 전달하는 데 초점이 맞추어져 있음
  - ✓ 1 계층의 주요 장비로는 허브(Hub), 리피터Repeater), 케이블(Cable), 커넥터(Connector), 트랜시버(Tranceiver), 탭(TAP)이 있음
  - ✓ 허브, 리피터는 네트워크 통신을 중재하는 네트워크 장비이고 케이블과 커넥터는 케이블 본체를 구성하는 요소이고 트랜시버는 컴퓨터의 랜카드와 케이블을 연결하는 장비이며 탭은 네트워크 모니터링과 패킷 분석을 위해 전기 신호를 다른 장비로 복제해주는 장비
  - ✓ 1 계층에서는 들어온 전기 신호를 그대로 잘 전달하는 것이 목적이므로 전기 신호가 1 계층 장비에 들어오면 이 전기 신호를 재생성하여 내보냄
  - ✓ 1 계층 장비는 주소의 개념이 없으므로 전기 신호가 들어온 포트를 제외하고 모든 포트에 같은 전기 신호를 전송

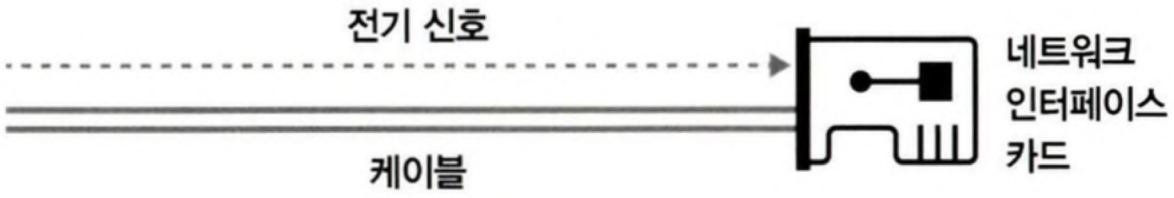
OSI 7계층 별 이해

- 2 계층(데이터 링크 계층)
  - ✓ 2 계층은 데이터 링크 계층으로 전기 신호를 모아 우리가 알아볼 수 있는 데이터 형태로 처리
  - ✓ 1 계층과는 다르게 전기 신호를 정확히 전달하기보다는 주소 정보를 정의하고 정확한 주소로 통신이 되도록 하는 데 초점이 맞추어져 있음
  - ✓ 1 계층에서는 전기 신호를 잘 보내는 것이 목적이므로 출발지와 목적지를 구분할 수 없지만 2 계층에서는 출발지와 도착지 주소를 확인하고 내게 보낸 것이 맞는지 또는 내가 처리해야 하는지에 대해 검사한 후 데이터 처리를 수행
  - ✓ 2 계층에서는 주소 체계가 생기면서 여러 통신이 한꺼번에 이루어지는 것을 구분하기 위한 기능을 주로 정의
  - ✓ 전기 신호를 모아 데이터 형태로 처리하므로 데이터에 대한 에러를 탐지하거나 고치는 역할을 수행할 수 있음
  - ✓ 신뢰할 수 있는 현대의 물리 계층과 달리 과거에는 신뢰할 수 없는 미디어를 이용해 통신하는 경우도 많아 2 계층에서 에러를 탐지하고 고치거나 재전송했지만 이더넷 기반 네트워크의 2 계층에서는 에러를 탐지하는 역할만 수행
  - ✓ 주소 체계가 생긴다는 의미는 하나의 장비와 만 통신하는 것이 아니라 동시에 여러 장비와 통신할 수 있다는 것이므로 무작정 데이터를 던지는 것이 아니라 받는 장비와 현재 데이터를 받을 수 있는지 확인하는 작업부터 해야 하는데 이 역할을 플로 컨트롤(Flow Control)이라고 부름

OSI 7계층 별 이해

❖ 2 계층(데이터 링크 계층)

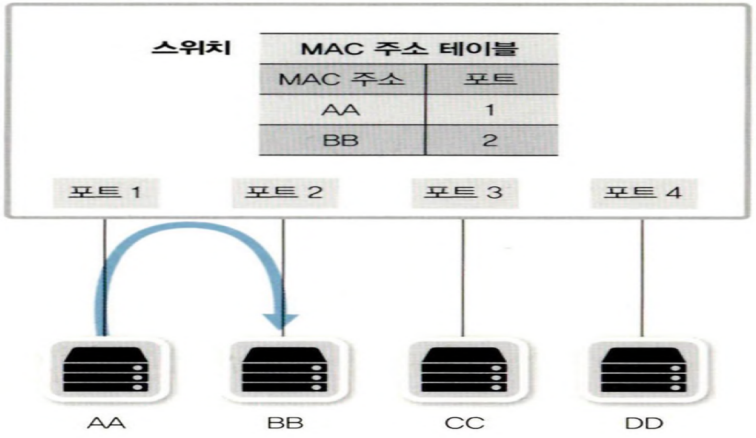
- ✓ 2 계층에서 동작하는 네트워크 구성 요소는 네트워크 인터페이스 카드(Network Interface Card) 와 스위치(Switch)
- ✓ 2 계층의 가장 중요한 특징은 MAC 주소라는 주소 체계가 있다는 것
- ✓ 2 계층에서 동작하는 네트워크 인터페이스 카드와 스위치 모두 MAC 주소를 이해할 수 있고 스위치는 MAC 주소를 보고 통신해야 할 포트를 지정해 내보내는 능력이 있음
- ✓ 네트워크 인터페이스 카드 동작 방식



- 전기 신호를 데이터 형태로 생성
- 목적지 MAC 주소와 출발지 MAC 주소를 확인
- 네트워크 인터페이스 카드의 MAC 주소를 확인
- 목적지 MAC 주소와 네트워크 인터페이스 카드가 갖고 있는 MAC 주소가 맞으면 데이터를 처리하고 다르면 데이터를 폐기

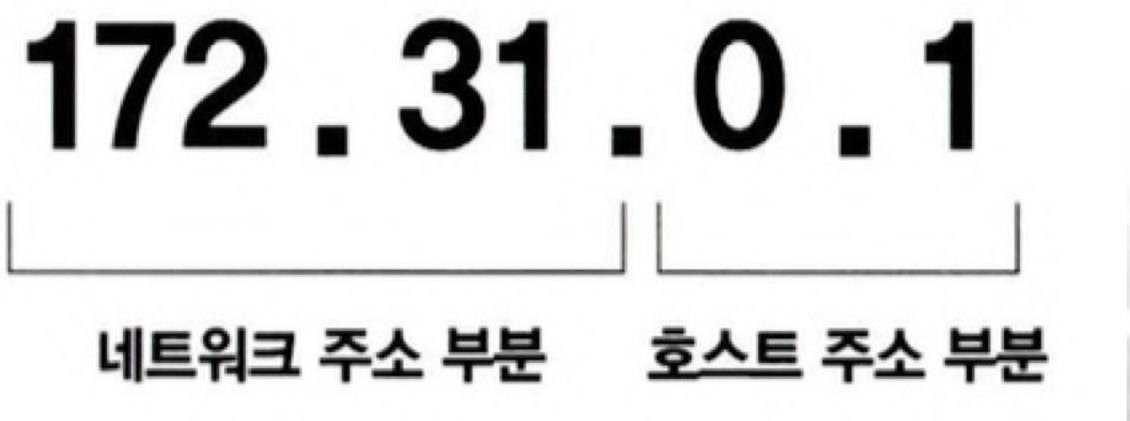
OSI 7계층 별 이해

- 2 계층(데이터 링크 계층)
  - ✓ 스위치 동작 방식
    - 스위치는 단말(Terminal)이 어떤 MAC 주소인지 연결된 포트는 어느 것인지 주소 학습(Address Learning) 과정에서 알 수 있음
    - 이 데이터를 기반으로 단말들이 통신할 때 포트를 적절히 필터링하고 정확한 포트로 포워딩해 줌
    - 1 계층에서 동작하는 허브는 한 포트에서 전기 신호가 들어오면 전체 포트로 전기 신호를 전달하다 보니 전체 네트워크에서 동시에 오직 하나의 장비만 데이터를 보낼 수 있음
    - 스위치의 적절한 필터링과 포워딩 기능으로 통신이 필요한 포트만 사용하고 네트워크 전체에 불필요한 처리가 감소하면서 이더넷 네트워크 효율성이 크게 향상되었고 이더넷 기반 네트워크가 급증하는 계기가 됨



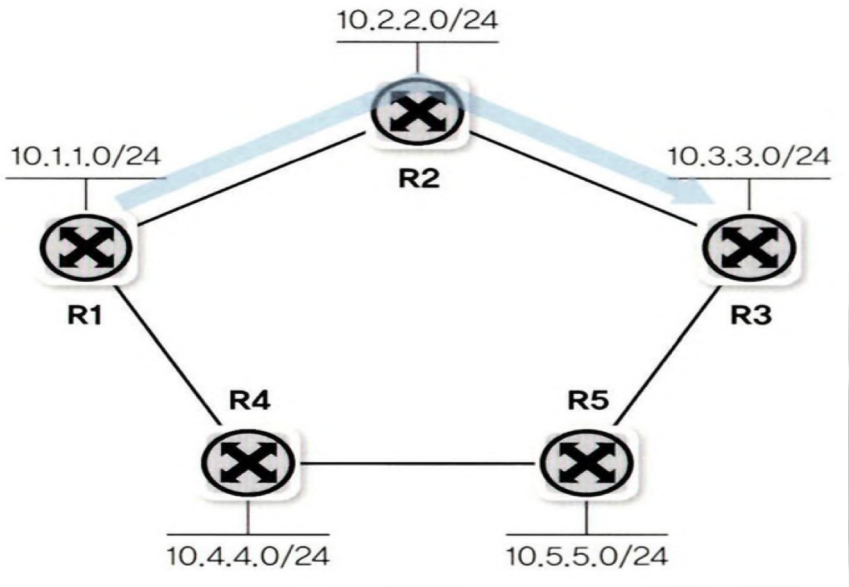
OSI 7계층 별 이해

- 3 계층(네트워크 계층)
  - ✓ 3 계층에서는 IP 주소와 같은 논리적인 주소를 정의
  - ✓ 데이터 통신을 할 때는 두 가지 주소가 사용되는데 2 계층의 물리적인 MAC 주소와 3 계층의 논리적인 IP 주소
  - ✓ IP 주소는 사용자가 환경에 맞게 변경해 사용할 수 있고 네트워크 주소 부분과 호스트 주소 부분으로 나눔



OSI 7계층 별 이해

- 3 계층(네트워크 계층)
  - ✓ 3 계층을 이해할 수 있는 장비나 단말은 네트워크 주소 정보를 이용해 자신이 속한 네트워크와 원격지 네트워크를 구분할 수 있고 원격지 네트워크를 가려면 어디로 가야 하는지 경로를 지정할 수 있음
  - ✓ 3 계층에서 동작하는 장비는 라우터인데 3 계층에서 정의한 IP 주소를 이해할 수 있음
  - ✓ 라우터는 IP 주소를 사용해 최적의 경로를 찾아주고 해당 경로로 패킷을 전송하는 역할을 수행



OSI 7계층 별 이해

- 4 계층(트랜스포트 계층)
  - ✓ 4 계층은 실제로 해당 데이터들이 정상적으로 잘 보내지도록 확인하는 역할을 수행
  - ✓ 패킷 네트워크는 데이터를 분할해 패킷에 실어보내다 보니 중간에 패킷이 유실되거나 순서가 바뀌는 경우가 생길 수 있는데 이 문제를 해결하기 위해 패킷이 유실되거나 순서가 바뀌었을 때 바로잡아 주는 역할을 4 계층에서 담당
  - ✓ 4 계층에서 패킷을 분할할 때 패킷 헤더에 보내는 순서와 받는 순서를 적어 통신하므로 패킷이 유실되면 재전송을 요청할 수 있고 순서가 뒤바뀌더라도 바로잡을 수 있음
  - ✓ 패킷에 보내는 순서를 명시한 것이 시퀀스 번호(Sequence Number)이고 받는 순서를 나타낸 것이 ACK 번호(Acknowledgement Number)
  - ✓ 장치 내의 많은 애플리케이션을 구분할 수 있도록 포트 번호(Port Number)를 사용해 상위 애플리케이션을 구분
  - ✓ 4 계층에서 동작하는 장비로는 로드 밸런서 와 방화벽이 있는데 4 계층에서 볼 수 있는 애플리케이션 구분자(포트 번호) 와 시퀀스, ACK 번호 정보를 이용해 부하를 분산하거나 보안 정책을 수립해 패킷을 통과 및 차단하는 기능을 수행

OSI 7계층 별 이해

- 5 계층(세션 계층)
  - ✓ 세션 계층(Session Layer)은 양 끝 단의 응용 프로세스가 연결을 성립하도록 도와주고 연결이 안정적으로 유지되도록 관리하고 작업 완료 후에는 이 연결을 끊는 역할을 수행
  - ✓ 세션을 관리하는 것이 주요 역할인 세션 계층은 TCP/IP 세션을 만들고 없애는 책임지고 에러로 중단된 통신에 대한 에러 복구와 재전송도 수행

OSI 7계층 별 이해

- 6 계층(프레젠테이션 계층)
  - ✓ 6 계층인 프레젠테이션 계층(Presentation Layer)은 표현 방식이 다른 애플리케이션이나 시스템 간의 통신을 돕기 위해 하나의 통일된 구문 형식으로 변환시키는 기능을 수행
  - ✓ 일종의 번역기나 변환기 역할을 수행하는 계층이고 이런 기능은 사용자 시스템의 응용 계층에서 데이터의 형식상 차이를 다루는 부담을 덜어줌
  - ✓ MIME 인코딩 이나 암호화, 압축, 코드 변환과 같은 동작이 이 계층에서 이루어 짐

OSI 7계층 별 이해

- 7 계층(애플리케이션 계층)
  - ✓ OSI 7 계층의 최상위 계층인 애플리케이션 계층(Application Layer)은 애플리케이션 프로세스를 정의하고 애플리케이션 서비스를 수행
  - ✓ 네트워크 소프트웨어의 UI 부분이나 사용자 입출력 부분을 정의하는 것이 애플리케이션 계층의 역할
  - ✓ 애플리케이션 계층의 프로토콜은 많은 종류가 있지만 대표적인 프로토콜로는 FTP, SMTP, HTTP, TELNET 이 있음

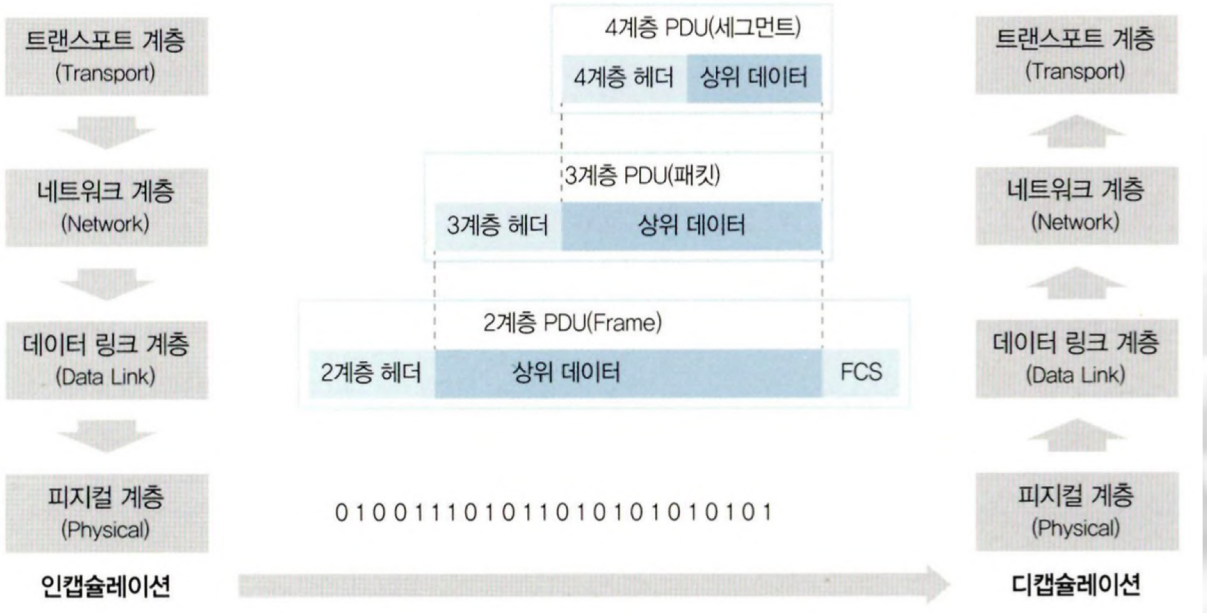
OSI 7계층 별 이해

❖ 계층 별 주요 프로토콜 및 장비

계층	주요 프로토콜	장비
애플리케이션 계층	HTTP, SMP, SMTP, STUN, TFTP, TELNET	ADC, NGFW, WAF
프레젠테이션 계층	TLS, AFP, SSH	
세션 계층	L2TP, PPTP, NFS, RPC, RTCP, SIP, SSH	
트랜스포트 계층	TCP, UDP, SCTP, DCCP, AH, AEP	로드 밸런서, 방화벽
네트워크 계층	ARP, IPv4, IPv6, NAT, IPSec, VRRP, 라우팅 프로토콜	라우터, L3 스위치
데이터 링크 계층	IEEE 802.2, FDDI	스위치, 브릿지, 네트워크 카드
피지컬 계층	RS-232, RS-449, V.35, S 등의 케이블	케이블, 허브, 탭(TAP)

Encapsulation & Decapsulation

- 개요
  - ✓ 상위 계층에서 하위 계층으로 데이터를 보내면 물리 계층에서 전기 신호 형태로 네트워크를 통해 신호를 보냄
  - ✓ 받는 쪽에서는 다시 하위 계층에서 상위 계층으로 데이터를 보냄
  - ✓ 데이터를 보내는 과정을 Encapsulation 이라고 하고 받는 과정을 Decapsulation 이라고 부름



Encapsulation & Decapsulation

- 패킷 기반 네트워크
  - ✓ 현대 네트워크는 대부분 패킷 기반 네트워크
  - ✓ 패킷 네트워크는 데이터를 패킷이라는 작은 단위로 쪼개 보내는데 이런 기법으로 하나의 통신이 회선 전체를 점유하지 않고 동시에 여러 단말이 통신하도록 함
  - ✓ 데이터를 패킷으로 쪼개고 네트워크를 이용해 목적지로 보내고 받는 쪽에서는 패킷을 다시 큰 데이터 형태로 결합해 사용
  - ✓ 데이터를 보내는 과정
    - 애플리케이션에서 데이터를 하위 계층으로 내려보내면서 패킷에 데이터를 넣을 수 있도록 분할하는데 이 과정을 Encapsulation이라고 부름
    - 네트워크 상황을 고려해 적절한 크기로 데이터를 쪼개고 4 계층부터 네트워크 전송을 위한 정보를 헤더에 붙여 넣음
    - 헤더 정보는 4 계층 3 계층 2 계층에서 각각 자신이 필요한 정보를 추가하는데 이 정보는 우리가 알아볼 수 있는 문자가 아닌 미리 정의된 비트 단위
    - 4 계층에서 헤더를 추가하고 3 계층으로 내려보내면 다시 3 계층에서 필요한 헤더 정보를 추가하고 2 계층으로 내려 보내고 2 계층에서도 다시 2 계층에서 필요한 헤더 정보를 추가한 후 전기 신호로 변환해 수신자에게 전송
    - 데이터를 전송하는 작업은 하위 계층에서만 3개의 헤더 정보가 추가됨

Encapsulation & Decapsulation

- ❖ 패킷 기반 네트워크
  - ✓ 데이터를 받는 과정
    - 받는 쪽에서는 Decapsulation 과정을 수행하는데 받은 전기 신호를 데이터 형태로 만들어 2 계층으로 전송
    - 2 계층에서는 송신자가 작성한 2 계층 헤더에 포함된 정보를 확인
    - 2 계층에 적힌 정보 중 목적지가 자신이 아니라면 자신에게 온 패킷이 아니므로 버리는데 랜 카드가 이 역할을 담당하며 2 계층에 적힌 정보의 목적지가 자신이 맞다면 3 계층으로 이 정보를 전송
    - 데이터를 상위 계층으로 전송할 때 2 계층의 헤더 정보는 더 이상 필요 없으므로 벗겨내고 전송
    - 이 데이터를 받은 3 계층에서는 2 계층이 동작했던 것처럼 상대방이 적은 3 계층의 헤더 정보를 확인해 자신에게 온 것이 맞는지 확인하고 맞으면 3 계층 헤더 정보를 제거하고 4 계층으로 보냄
    - 이를 받은 4 계층도 3 계층과 같은 과정을 거쳐 데이터를 애플리케이션에 전송

Encapsulation & Decapsulation

❖ 잘 알려진 상위 프로토콜 지시자: 포트번호

포트 번호	서비스
TCP 20, 21	FTP(File Transfer Protocol)
TCP 22	SSH(Secure Shell)
TCP 23	TELNET(Telnet Terminal)
TCP 25	SMTP(Simple Mail Transport Protocol)
UDP 49	TACACS
TCP 53/UDP 53	DNS(Domain Name Service)
UDP 67, 68	BOOTP(Bootstrap Protocol)
TCP 80/UDP 80	HTTP(HyperText Transfer Protocol)
UDP 123	NTP(Network Time Protocol)
UDP 161, 162	SNMP(Simple Network Management Protocol)
TCP 443	HTTPS
TCP 445/UDP 445	Microsoft-DS

Encapsulation & Decapsulation

- MSS & MTU
  - ✓ 애플리케이션에서 데이터를 만들어 보낼 때 하위 계층에서 네트워크 상황에 맞게 데이터를 잘 쪼개 상대방에게 전달
  - ✓ 네트워크에서 수용할 수 있는 크기를 역산정해 데이터가 4 계층으로 내려올 때 적절한 크기로 쪼개질 수 있도록 유도하는데 이 값을 MSS(Maximum Segment Size) 라고 부름
  - ✓ 네트워크에서 한 번에 보낼 수 있는 데이터 크기를 MTU(Maximum Transmission Unit) 라고 부르며 일반적인 이더넷에서 수용할 수 있는 크기는 1,500바이트
  - ✓ MTU 와 MSS는 모두 데이터 크기를 지칭하는 것이므로 MTU값은 2 계층의 데이터 크기이고 MSS는 4 계층에서 가질 수 있는 최대 데이터 크기
  - ✓ 2 계층에서는 2 계층 헤더들의 크기를 제외한 데이터 크기를 MTU 크기라고 부름
  - ✓ IP 헤더와 TCP 헤더의 표준 헤더 크기는 일반적으로 각각 20바이트이므로 일반 이더넷인 경우 MSS 값을 1,460바이트로 사용