

네트워크 기초

TCP / IP

게임공학부 게임공학과

2017180035 장수현

화 2-3 목 4 교시

페이지 0 / 15

[목차]

1. TCP/IP란?

- TCP/IP의 정의
- TCP/IP의 역사

2. TCP/IP의 계층 구조

- OSI 7 계층구조
- TCP/IP 계층구조
- TCP/IP 계층에 의한 데이터 전송

3. TCP와 UDP

- TCP
- UDP

4. IP Address

- Subnet Mask
- Class A
- Class B
- Class C
- IPv4 VS IPv6

5. 참고문헌 및 출처

TCP/IP의 정의

TCP/IP는 TCP라는 Transmission Control Protocol과 IP라는 Internet Protocol의 약자를 합쳐서 부르는 말이다. 네트워크 전송 프로토콜로써 인터넷을 통해 데이터를 전송하고 받을 때 필요한 프로토콜을 말한다. 프로토콜이라는 것은 데이터 통신을 원활히 하기 위한 규칙들의 집합을 말한다. 프로토콜에는 3가지 주요요소인 구문(Syntax), 의미(Semantics), 타이밍(Timing)이 있다.

- 구문(Syntax): 데이터 구조나 형식을 말하는 것으로 데이터가 어떠한 순서로 나타나는지를 의미한다.
- 의미(Semantics): 비트들의 각 부분의 의미를 말한다. 특정 패턴을 어떻게 해석하며, 이를 기반으로 어떤 동작을 할 것인가 결정하는 영역이다. 예를 들면 주소는 메시지의 최종 목적지나 선택되는 경로를 나타내게 된다.
- 타이밍(Timing): 타이밍에는 두가지 특성이 있다. 첫번째로는 언제 데이터를 전송할 것인가이다. 두번째로는 얼마나 빨리 전송할 것인가이다. 예를 들면 보내는 사람(송신자)이 10Mbps로 데이터를 보내고 받는 사람(수신자)은 1Mbps로 데이터를 처리한다면 상대적으로 많은 양의 데이터가 전송되어 많은 데이터가 손실될 것이다.

TCP/IP의 역사

1960년대 컴퓨터 시스템들은 시스템만의 프로토콜을 사용하여서 다른 기종 간에 연결이 불가능했다. 이러한 문제점을 해결하기 위해서 개발된 것이 ARPANET(Advanced Research Projects Agency Net)이다. ARPANET은 해당 네트워크가 속한 한 그룹이 폭파하더라도 전체 네트워크가 붕괴되지 않는 구조로 설계되었다. 이 이유로 많은 업체들이 ARPANET을 지원하게 되었으며 이것이 인터넷의 시초이다. ARPANET이 발전되면서 서로 다른 물리적 구조를 가진 네트워크 간에도 통신을 할 수 있도록 하는 절대적 요구가 발생하였으며 이 요구를 해결하기 위한 방법으로 TCP/IP가 개발되었다.

OSI 7 계층구조

OSI 7 계층은 네트워크 통신을 이해하기 위한 기초이다. OSI 7계층을 이해하는 것은 TCP/IP를 이해하는데 도움이 된다. OSI 7 계층은 “국제 표준화 기구(ISO)에서 개발한 모델로, 컴퓨터나 네트워크 프로토콜 디자인과 통신을 계층으로 나누어 설명한 것이다”

[\[https://ko.wikipedia.org/wiki/OSI_%EB%AA%A8%ED%98%95\]](https://ko.wikipedia.org/wiki/OSI_%EB%AA%A8%ED%98%95)

OSI 7 계층은 기능을 기준으로 하여서 계층을 나누었다. 하위 계층은 하드웨어로, 상위 계층은 소프트웨어로 구성된다. 각 계층은 자신의 하위 계층의 기능을 이용하고 상위 계층에게는 기능을 제공한다. 소제목에서도 알 수 있듯이 계층은 7개로 구성되어 있다.

- **1 Layer: Physical Layer**

실제 장치들을 연결하기 위한 전기적, 물리적 특징을 이용하여 데이터를 전송한다. 이 계층에서는 데이터 전달만 이루어진다. "데이터가 무엇인지, 어떤 에러가 있는지 등에는 전혀 신경 쓰지 않는다." [<https://shlee0882.tistory.com/110>] 기기간의 비트의 전송으로 이루어진다.

- **2 Layer: Data Link Layer**

정보의 전달을 할 수 있도록 도와주는 역할이다. bit단위의 정보를 frame단위의 정보로 바뀌어서 네트워크 계층으로 보내거나 frame단위의 정보를 bit단위로 쪼개서 물리 계층으로 보낸다. (L1 → L2 or L2 → L1) 물리계층에서 발생할 수 있는 오류를 찾고, 수정을 한다.

Node address를 정의한다. Node Address는 네트워크 방지에 부여된 고유 식별 번호이고 Mac Address라고 한다.

- **3 Layer: Network Layer**

"다중 네트워크 링크에서 패킷을 발신지로부터 목적지로 전달할 책임을 갖는다."

[<http://blog.naver.com/PostView.nhn?blogId=demonicws&logNo=40117378644>]

최종 목적지까지 가장 최적의 경로를 찾는다. 논리적인 주소 구조를 갖으며 이걸 IP Address라고한다.

- **4 Layer: Transport Layer**

Port Number를 가지고 있다. Port Number는 네트워크 메시지가 도착했을 때 전달되어야 하는 특정 프로세스를 인식하기 위한 방법이다.

- **5 Layer: Session Layer**

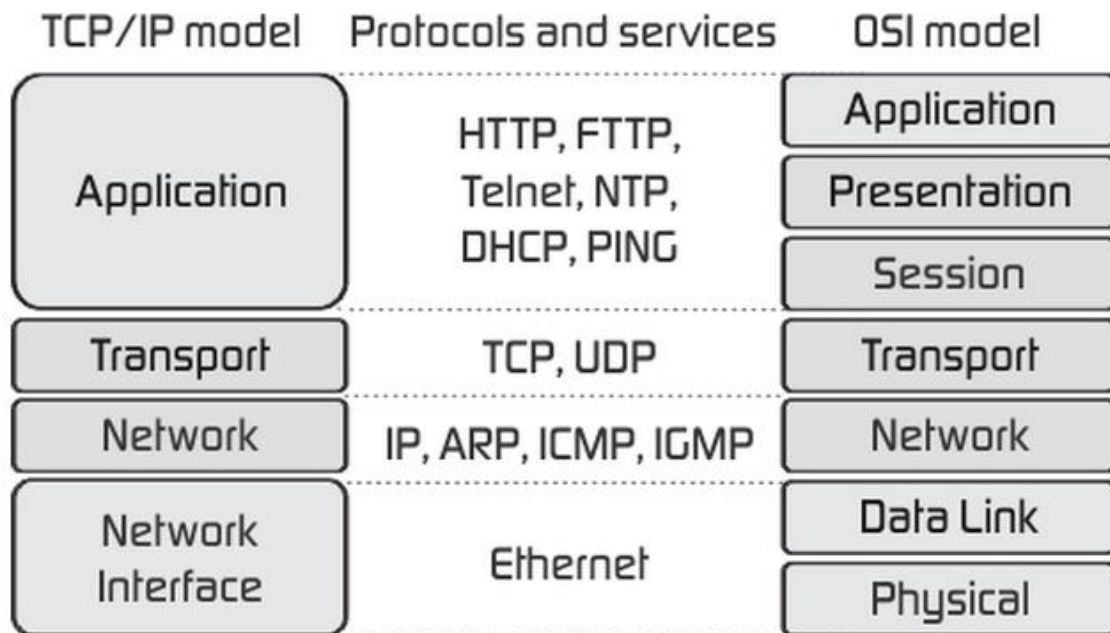
두 장치 사이의 접속 연결이나 동기화를 하는 역할을 한다. 사용자 간의 port 연결이 유효한지 확인하는 역할을 한다.

- **6 Layer: Presentation Layer**

인코딩이나 암호화하는 역할을 한다. 데이터를 동일한 형태로 바꿔주는 역할을 한다. 이를 통해서 두 장치가 데이터를 이해할 수 있도록 하는 역할을 한다.

- **7 Layer: Application Layer**

사용자가 네트워크에 접근할 수 있도록 해준다. 사용자에게 인터페이스와 같은 역할을 한다. file, print, email같은 서비스를 제공한다. “응용 프로세스들 사이의 전환을 제공한다.”



[https://ko.wikipedia.org/wiki/OSI_%EB%AA%A8%ED%98%95]

[그림 1] TCP/IP 모델과 OSI 모델 계층

TCP/IP 계층구조

OSI 7계층과 달리 TCP/IP 계층은 4계층으로 이루어져 있다. TCP/IP 계층을 5개로 나누는 경우도 있다. TCP/IP의 각 계층은 OSI 7계층의 1~2개에서 2~3개의 계층에 해당하는 역할을 수행한다. TCP/IP 모델은 정식 표준은 아니지만 인터넷이 유명해지면서 같이 떠오르게 됐다.

- 1 Layer: Network Interface(access) Layer

네트워크 인터페이스 계층은 OSI 7계층에서 physical 계층과 data link 계층에 해당하는 역할을 수행한다. 이 계층은 데이터 통신할 때 TCP/IP 패킷들을 전달하고 받는 과정을 담당한다. 데이터의 단위를 맞는 데이터 형식으로 바꾸거나 네트워크 접근시에 신호를 결정한다. 이 계층에서는 고유 식별 번호인 MAC Address와 CSMA/CD, Ethernet을 사용한다.

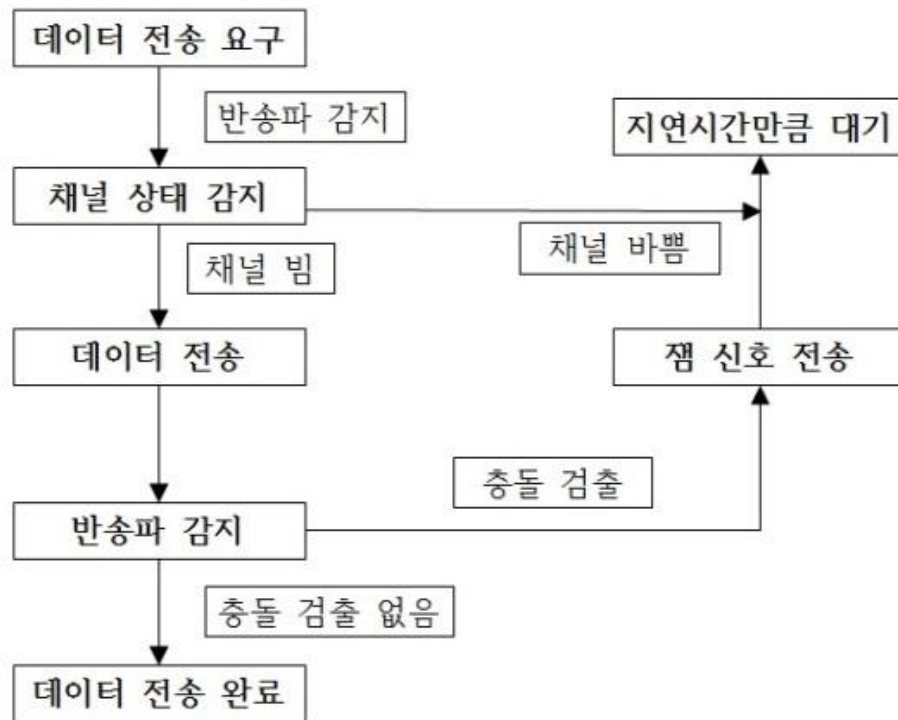
CSMA/CD는 Carrier Sense Multiple Access / Collision Detection의 약자로 Ethernet에서 사용되는 말이다. CSMA/CD는 “전류의 강도를 통해 케이블이 사용 중인지를 확인하고 트래픽이 없으면 패킷을 송출하는 LAN의 매체 점유 방법 중 한가지 방법”이다.

[<https://needjarvis.tistory.com/156>] CSMA/CD는 CS, MA, CD 세가지로 설명할 수 있다.

CS(Carrier Sense)는 데이터를 보내기 전에 데이터가 이동하고 있는가를 감지 한다.

MA(Multiple Access)는 CS작업 후에 이동하는 데이터가 없으면 데이터를 전송한다.

CS(Collision Detection)는 CMA 후에 다른 네트워크에서도 같은 작업을 하고 MA에서 시간 간격이 거의 없이 전송했을 때 데이터 충돌이 일어나는지를 감지하고 알려준다.



[그림 2] CSMA/CD의 동작 원리

- 2 Layer: Internet(Network) Layer

인터넷 계층은 OSI 7계층에서 네트워크 계층에 해당하는 역할을 수행한다. IP Address를 통해서 패킷을 전송하는 기능과 라우팅(Routing) 기능을 담당한다. 라우팅이란 패킷 스위칭 네트워크에서 패킷이 목적지로 가는 가장 효율적인 경로를 정해주는 역할을 말한다. 인터넷 계층에는 IP, ARP, ICMP, RARP로 구성되어 있다.

IP(Internet Protocol)는 인터넷에서 사용하는 프로토콜이다. 송수신 컴퓨터 사이의 데이터 묶음을 보내는 경로를 찾아주는 프로토콜이다. IP는 뒤에서 더 설명을 하겠다.

ARP(Address Resolution Protocol)는 데이터를 받는 대상의 MAC address를 모를 때 IP 주소를 이용하여 MAC Address를 요청하는 프로토콜이다.

RARP(Reverse Address Resolution Protocol)는 ARP와 반대로 자신의 MAC Address를 알지만 IP 주소를 모르는 경우 IP Address를 요청하는 프로토콜이다.

ICMP(Internet Control Message Protocol)는 오류나 흐름의 제어를 보완하기 위한 프로토콜이다. "인터넷 프로토콜에 의존하여 작업을 수행한다."

{ https://ko.wikipedia.org/wiki/%EC%9D%B8%ED%84%B0%EB%84%B7_%EC%A0%9C%EC%96%B4_%EB%A9%94%EC%8B%9C%EC%A7%80_%ED%94%84%EB%A1%9C%ED%86%A0%EC%BD%9C}

- 3 Layer: Transport Layer

통신 계층은 OSI 7계층에서 통신 계층에 해당하는 역할을 수행한다. 주로 송수신자 사이의 통신 노드 간의 연결을 제어하며 신뢰성 있는 데이터 전송 기능을 제공한다.

TCP(Transmission Control Protocol)와 UDP(User Datagram Protocol) 프로토콜을 사용한다.

TCP는 연결 지향 프로토콜이고 UDP는 비 연결 지향 프로토콜이다. TCP는 UDP에 비해 속도는 느리지만 정확성이 높다. 이에 반해 UDP는 속도는 빠르지만 정확성은 높지 않다.

TCP와 UDP는 뒤에서 더 자세하게 설명하겠다.

- 4 Layer: Application Layer

응용 계층은 OSI 7계층에서 세션 계층, 표현 계층, 응용 계층이 합쳐서 역할을 수행한다. 응용 계층에서는 데이터의 형식을 정하고 부호화, 암호화하는 작업을 한다. 응용 계층의 TCP, UDP기반의 서비스를 구현할 때 사용한다. 대표적으로 HTTP, FTP, Telnet, DNS(Domain Name Service), SMTP가 있다.

HTTP(Hyper Text Transfer Protocol)는 "www상에서 정보를 주고받을 수 있는 프로토콜이다. 주로 HTML 문서를 주고받는 데에 쓰인다. TCP와 UDP를 사용"한다.

[<https://ko.wikipedia.org/wiki/HTTP>] 클라이언트가 HTTP를 통하여 서버로부터 데이터를 요청하면 서버는 요청에 필요한 HTTP를 통해 사용자에게 정보를 알려준다. http: 라는 URL주소로 조회할 수 있다.

FTP(File Transfer Protocol)는 "TCP/IP 프로토콜을 가지고 서버와 클라이언트 사이의 파일 전송을 하기 위한 프로토콜이다." [https://ko.wikipedia.org/wiki/파일_전송_프로토콜] 역사가 오래 되었지만 지금까지도 인터넷에서 많이 사용하고 있다.

Telnet(Terminal Emulation Protocol)는 인터넷을 통하여 원격지의 호스트 컴퓨터에 접속하기 위해 사용되는 프로토콜이다. 문자기반의 원격 가상 단말 기능을 하는 인터페이스이다. 정보는 문자나 줄 단위로 정보가 송수신된다.

DNS(Domain Name Service)는 도메인을 컴퓨터가 읽을 수 있는 IP주소로 바꾸는 시스템

이다. IP 주소를 기억하는 것이 어려워 Jon Postel과 Paul Mockapetris에 의해 나온 방식이다. 도메인을 입력하면 DNS 서버에서 IP Address로 바꿔 전달하여 얻는 방식이다.

SMTP(Simple Mail Transfer Protocol)는 “인터넷에서 이메일을 보내기 위해 이용되는 프로토콜이다.” “메일 서버 간의 송수신뿐만 아니라, 메일 클라이언트에서 메일 서버로 메일을 보낼 때에도 사용되는 경우가 많다.”[https://ko.wikipedia.org/wiki/간이_우편_전송_프로토콜]

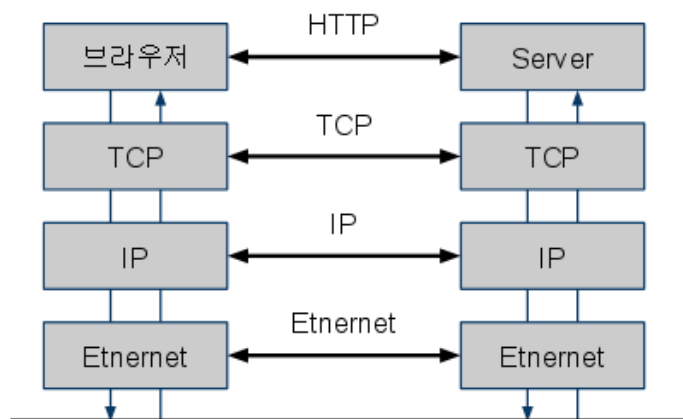
TCP/IP 계층에 의한 데이터 전송

TCP/IP 계층에 의한 데이터 전송에 대해 WWW로 예를 들어 얘기해보자. WWW는 HTTP라는 프로토콜을 사용한다. 사용자는 브라우저를 사용하여 URL 입력을 통해서 웹페이지를 요청한다. 사용자의 요청은 전달하기 용의한 TCP 패킷으로 만들어진다. 이것은 다시 원하는 주소로 이동할 수 있도록 IP 패킷으로 다시 만들고 Ethernet 카드로 보내서 Internet으로 나가게 된다.

원하는 주소로 TCP/IP 패킷을 보내기 위한 여러 장치들을 이용하여 Ethernet 카드로 TCP/IP 패킷이 전달되게 된다. Ethernet 카드는 TCP/IP 패킷을 Internet layer로 보낸다. 여기서는 IP 패킷을 분석하여 어디서 왔으며, 어디로 가는지 판단하게 된다.

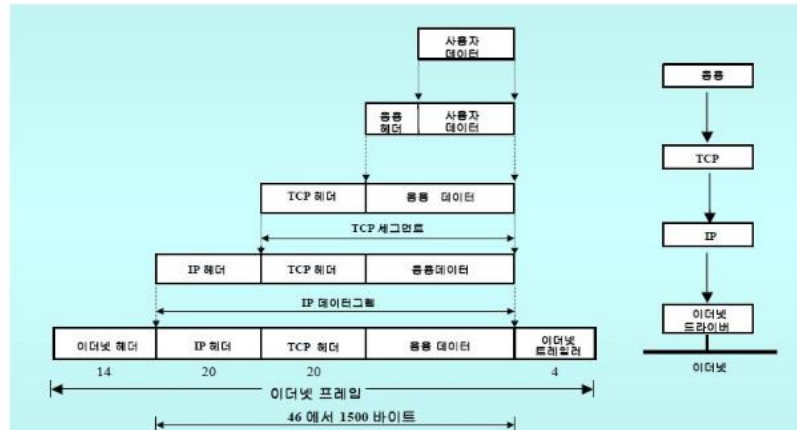
목적지가 자신이면 다시 Transport layer로 보내고, TCP 프로토콜을 사용하여 누락 메시지가 있다면 다시 요청하여 순서를 재조합 하는 등 메시지를 검사해서 다시 Application layer로 보낸다.

Application layer에서는 웹서버가 HTTP 프로토콜에 맞게 검사를 하고 사용자가 요청한 웹페이지를 읽어서 transport layer로 보낸다. 웹페이지를 브라우저까지 보내는 과정은 이 과정을 반대로 따라가면 된다. 브라우저는 웹페이지를 받아서 HTTP를 기반으로 렌더링 작업을 한 후에 화면에 뿌린다. 각 계층은 대응하는 상대의 계층만을 상관하는 계층적 구조를 가진다.



[그림 3] TCP/IP Packet Internet

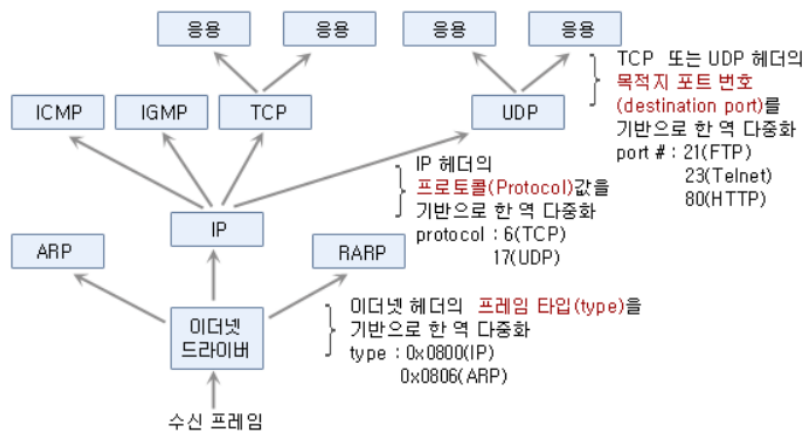
캡슐화 과정



캡슐화(Encapsulation): 각 계층에서 해야 하는 일을 헤더에 붙이는 작업

[\[http://www.hoons.net/board/cshaptip/content/35343\]](http://www.hoons.net/board/cshaptip/content/35343)

[그림 4] TCP/IP 데이터 캡슐화



역 다중화: Ethernet에서 넘어온 데이터의 헤더를 파악하여 필요한 부분으로 쪼개서 보내는 작업

[그림 5] TCP/IP 역 다중화

TCP와 UDP

TCP와 UDP는 TCP 계층에서 3번째 통신 계층에 해당한다. 데이터를 보내기 위해 사용하는 프로토콜이다.

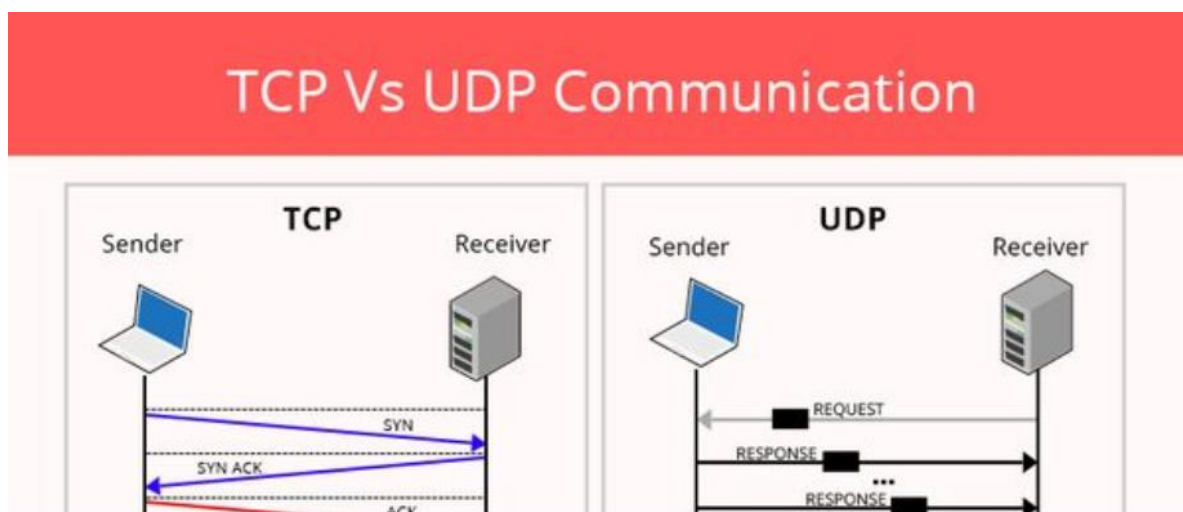
TCP

TCP는 앞서서도 말했듯이 연결 지향 프로토콜이다. 연결 프로토콜은 데이터 전송 시에 1대1로 송수신자가 모두 연결이 됐는가 확인 후에 데이터 전송이 된다. 신뢰성을 중요하게 여기기 때문이

다. 수신자 버퍼의 오버 플로우 현상을 방지해주고 네트워크 내 패킷 수가 과도하게 증가하는 현상을 제어해주는 등 데이터의 간섭이나 손실을 검색하여 보정하고, 순서를 재조합 한다. 이러한 과정들로 인해 TCP는 UDP보다는 속도가 떨어지지만 신뢰성은 올라가게 된다.

UDP

UDP는 TCP와 반대로 비연결 지향 프로토콜이다. 연결 지향 프로토콜과는 다르게 송수신자가 모두 연결되었는가를 확인한 후에 데이터를 보내는 것이 아니라 주소를 가지고 목적지로 전달만 한다. 일방적으로 데이터를 전달하는 방식이다. 전송만 하기 때문에 손실된 데이터의 정보를 알 수가 없다. UDP는 TCP에 비해 신뢰성이 떨어지는 프로토콜이지만 빠르게 데이터를 전송할 수 있다.



[그림6] TCP와 UDP 소통 방식

IP Address

IP란 Internet Protocol의 약자로 인터넷에서 사용하는 프로토콜이다. TCP 계층에서 두번째 계층인 인터넷 계층에 해당한다. IP 및 IP Address는 인터넷 계층에서 아주 중요하다. IP는 호스트 주소 표기, 패킷 분할에 관한 기능을 제공한다. 하지만 “End-To-End 형식의 오류 제어나 흐름제어 기능은 제공하지 않는다.” [\[https://dany-it.tistory.com/50\]](https://dany-it.tistory.com/50)

IP는 비신뢰성과 비연결성의 특징을 가지고 있다. 비연결성은 UDP와 같이 연결되었는가를 확인하지 않고 데이터가 있으면 전송하는 것이다. “신뢰성은 흐름에 관여하지 않기 때문에 보낸 정보가 제대로 갔는지 보장하지 않는다.” [\[https://ko.wikipedia.org/wiki/인터넷_프로토콜\]](https://ko.wikipedia.org/wiki/인터넷_프로토콜)

패킷을 분할, 병합하는 기능을 수행한다. 상위 계층에서 내려온 데이터를 받아서 처리할 때 프로토콜의 크기가 다르다면 데이터를 패킷의 크기에 맞게 쪼개서 나눈다.

데이터 체크섬은 제공하지 않고, 헤더 체크섬만 제공한다. 여기서 체크섬이란 “간단하게 에러 검

출을 하는 방법"이다.

[http://www.ktword.co.kr/abbr_view.php?nav=&m_temp1=1477&id=744]

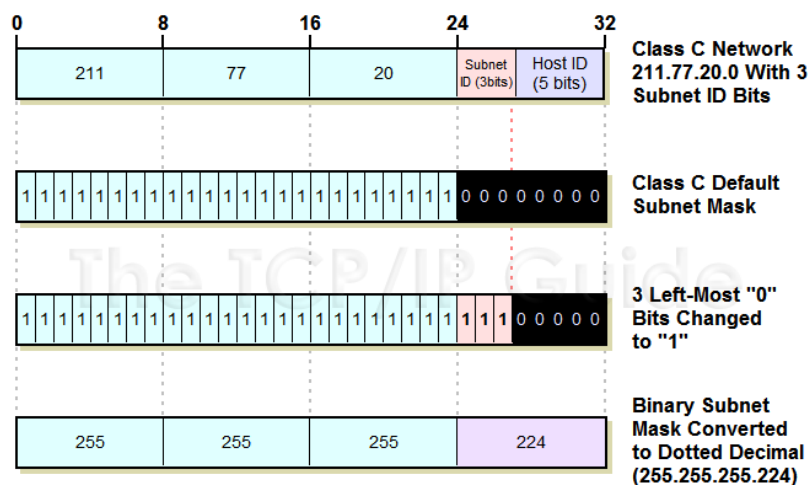
Best Effort 원칙에 따른 전송 기능을 제공한다. Best Effort는 말 그대로 최선의 노력을 하는 것을 말한다. 데이터그램의 전송을 위해서 최대의 노력을 하지만 확실한 전송을 보장할 수 없다.

IP Address는 32Bit로 구성되어 있다. 사람은 32비트 주소를 외우기 힘들기 때문에 응용계층에 속하는 DNS 프로토콜을 이용하여 IP Address를 사용자가 보기 편하게 바꿀 수 있다. IP Address는 Class A, B, C로 나누어진다. 세 클래스를 구별할 때 서브넷 마스크(Subnet Mask)라는 것을 사용한다.

Subnet Mask

서브넷 마스크는 IP Address를 좀 더 효율적으로 관리하기 위해서 나온 개념이다. 네트워크의 범위를 할당 받은 IP를 용도와 필요에 따라 분리해 전송방식을 결정하기 위해 생겼다.

IP Address의 번호 부분을 모두 0으로 초기화한 후에 클래스는 나타내는 부분은 유지하기 위해 1로 초기화해서 IP Address와 AND 연산을 통해 같은 네트워크인지 구분한다.



[그림 7] Class C로 Subnet Mask 연산의 예

Class A

Class A는 1~126 범위의 IP주소를 가진다. 나머지 두번째, 세번째, 네번째 단위는 자유롭게 네트워크 사용자에게 부여할 수 있는 IP이다. 여기서 0, 127은 예약된 범위이다. 최고 위의 class이다.

Class B

Class B는 128~191 범위의 IP주소를 가진다. 두번째 단위의 숫자는 접속할 수 있는 네트워크를 말한다. 두번째로 높은 단위의 class이다.

Class C

Class c는 192~223 범위의 IP주소를 가진다. 두번째와 세번째 단위의 숫자는 접속할 수 있는 네트워크를 말하며 마지막 네번째 단위의 254개는 네트워크 사용자에게 자유롭게 부여할 수 있는 IP이다. 두 개는 예약된 범위이다.

IPv4 VS IPv6

Class A, B, C는 IPv4에 속한다. 인터넷은 원래 IPv4 프로토콜로 구축되어 있었으나 32 bit라는 제한된 주소라는 한계가 있다. 인터넷이 지속적으로 발전해오면서 부족한 주소를 해결하기 위해서 IPv6 프로토콜이 제안되었다. IPv4는 32bit로 4Byte였지만 IPv6는 128bit인 16Byte로 4배가 늘었다. IP Address는 2^{32} 에서 2^{128} 로 늘어났다. IPv6는 주소공간을 다음과 같이 16비트를 16진수로 표현하여 8자리로 나타낸다.

```
2001:0DB8:0000:0000:0000:0000:1428:57ab
2001:0DB8:0000:0000:0000::1428:57ab
2001:0DB8:0:0:0:0:1428:57ab
2001:0DB8:0::0:1428:57ab
2001:0DB8::1428:57ab
```

[그림 8] IPv6의 주소공간 표현 예

■IPv6와 IPv4 프로토콜 비교

구분	IPv4	IPv6
주소체계	32bit	128bit
표기법	8bit, 4부분, 10진수 표기 예) 127.0.0.1	16bit, 8부분, 16진수 표기 예) 2010:0101: ... :6ABC
주소개수	43억 개	(43억) ⁴ 개
할당방법	A,B,C 클래스 비순차 할당	N/W규모별 순차할당
QoS	어려움(Best effort)	Traffic class, Flow Level
보안	별도 구현	IPsec 내장
P&P	N/A	지원
모바일IP	어려움	지원(IP무제한)
웹캐스팅	어려움	지원(Scope Field 증가)

[그림 9] IPv4 vs IPv6

■ IPv6와 IPv4 주소대응체계 비교

구분	IPv4	IPv6
멀티캐스트	224.0.0.0 / 4 (D클래스)	FF00 :: / 8
브로드캐스트	255.255.255.255	N/A
공인 IP	공인 IP 별도 신청	Global Unicast
사설 IP	10, 172, 192 3개 영역	N/A
미지정주소	0.0.0.0 / 32	:: / 128
루프백주소	127.0.0.1	:: 1 / 128
링크로컬주소	169.254.0.0 / 16	FE80 :: / 64

[그림 10] IPv4 vs IPv6 - 주소대응체계

참고문헌 및 출처

[정석용의 TCP / IP 소켓 프로그래밍 - 저자: 정석용] - 프리렉(2009)]

[TCP / IP 프로토콜 - Behrouz A. Forouzan 원 저] - MRC 미래컴

[강의 자료 - Chapter 1 Data Communication - An Introduction]

<그림>

[그림 1] <http://fiberbit.com.tw/tcpip-model-vs-osi-model/>

[그림 2]

<https://m.blog.naver.com/PostView.nhn?blogId=stop2y&logNo=100209824948&proxyReferer=https%3A%2F%2Fwww.google.com%2F>

[그림 3]

<https://soofi.tistory.com/entry/%EC%BA%A1%EC%8A%90%ED%99%94-%EC%97%AD%EC%BA%A1%EC%8A%90%ED%99%94>

[그림 5] <https://redreans.tistory.com/106>

[그림 6] <https://www.colocationamerica.com/blog/tcp-ip-vs-udp>

[그림 7]

http://www.tcpipguide.com/free/t_IPSubnettingStep3DeterminingTheCustomSubnetMask.htm

[그림 8] <https://ko.wikipedia.org/wiki/IPv6>

<https://breath91.tistory.com/entry/%ED%94%84%EB%A1%9C%ED%86%A0%EC%BD%9C%EC%9D%98-%EC%A3%BC%EC%9A%94%EC%9A%94%EC%86%8C-3%EA%B0%80%EC%A7%80>

<https://netlog.tistory.com/entry/TCPIP>

https://ko.wikipedia.org/wiki/OSI_%EB%AA%A8%ED%98%95

<https://namu.wiki/w/OSI%20%EB%AA%A8%ED%98%95?from=OSI>

<http://www.terms.co.kr/portnumber.htm>

<https://hahahoho5915.tistory.com/15>

https://ko.wikipedia.org/wiki/%EB%B0%98%EC%86%A1%ED%8C%8C_%EA%B0%90%EC%A7%80_%EB%8B%A4%EC%A4%91_%EC%A0%91%EC%86%8D_%EB%B0%8F_%EC%B6%A9%EB%8F%8C_%ED%83%90%EC%A7%80

<https://m.blog.naver.com/PostView.nhn?blogId=stop2y&logNo=100209824948&proxyReferer=https%3A%2F%2Fwww.google.com%2F>

https://ko.wikipedia.org/wiki/%EC%A3%BC%EC%86%8C_%EA%B2%B0%EC%A0%95_%ED%94%84%EB%A1%9C%ED%86%A0%EC%BD%9C

https://ko.wikipedia.org/wiki/%EC%97%AD%EC%88%9C_%EC%A3%BC%EC%86%8C_%EA%B2%B0%EC%A0%95_%ED%94%84%EB%A1%9C%ED%86%A0%EC%BD%9C

https://ko.wikipedia.org/wiki/%EC%9D%B8%ED%84%B0%EB%84%B7_%EC%A0%9C%EC%96%B4_%EB%A9%94%EC%8B%9C%EC%A7%80_%ED%94%84%EB%A1%9C%ED%86%A0%EC%BD%9C

<https://hahahoho5915.tistory.com/13>

<https://ko.wikipedia.org/wiki/HTTP>

<https://namu.wiki/w/%ED%85%94%EB%84%B7>

<https://aws.amazon.com/ko/route53/what-is-dns/>

<http://www.hoons.net/board/cshaptip/content/35343>

<https://rhyshan.com/286>

<https://ko.wikipedia.org/wiki/IPv4>

<https://ko.wikipedia.org/wiki/IPv6>

<http://blog.naver.com/PostView.nhn?blogId=demonicws&logNo=40117378644>