☞ IP 주소란?

네트워크상에 존재하는 컴퓨터들을 구분하고, 서로를 인식하기 위해 사용하는 특수한 번호.
32-bit 체계의 IPv4와, 128-bit 체계의 IPv6가 있다.

☞ About IPv4

32-bit의 길이로 이루어지는 IPv4는 **1 byte (= 8-bit)씩 4개로 나누어 10진수로 표현**한다.
또한, 나누어진 **4개의 10진수는 마침표(.)로 구분**한다.

네트워크를 구분하기 위한 "네트워크 식별자(NetID / Network Identifier)"와 네트워크 내에 있는 호스트들을 구분하기 위한 "호스트 식별자(HostID / Host Identifier)"로 구성되며, 5개의 Class로 구분된다.

Network Bit			Host Bit
192	168	0	1
11000000	10101000	00000000	00000001

IPv4 주소는 [A, B, C, D, E] 5개의 Class로 분류되며, [A, B, C] Class만이 Network/Host 주소 체계를 갖는다.

@ Class별 구조

A Class	8-bit		24-bit
	0	Network ID	Host ID
최상위 bit가 "0"으로 시작하며, [0 ~ 127]로 시작하는 IP 주소.			

B Class	8-bit		24-bit
	1 0	Network ID	Host ID
최상위 bit가 "10"으로 시작하며, [128 ~ 191]로 시작하는 IP 주소.			

C Class	24-bit		8-bit
	1 0	Network ID	Host ID
최상위 bit가 "110"으로 시작하며, [192 ~ 223]로 시작하는 IP 주소.			

D Class	1 1 1 0	MultiCast 주소
최상위 bit가 "1110"으로 시작하며, MultiCast 용도로 사용하기 위한 IP 주소.		

E Class	1 1 1 1	예약 주소 (Reserved)
최상위 bit가 "1111"으로 시작하며, 특수 연구 용도로 사용하는 IP 주소.		

@ 각 Class별 주소의 범위

	시작 Bit	첫 byte 10진 주소 범위	최대 Subnet 수	Subnet 당 최대 Host 수
A Class	0	0 ~ 127	127	16,777,214
B Class	10	128 ~ 191	16,384	65,535
C Class	110	192 ~ 223	2,097,152	254
D Class	1110	224 ~ 239	-	-
E Class	1111	240 ~ 255	-	-

☛ IPv4의 사설 IP(Private IP)

부족한 공인 IP 문제를 해결하기 위해 사용하는 내부적인 IP.
공인 IP와는 달리 **인터넷에서 Routing이 불가능**하며, **동일 network 내부에서만 인식**이 되고 다른 network에 접속 할 수 없다.
외부 인터넷에 연결하는 것이 불가능하므로 내부의 수많은 컴퓨터들이 외부의 공인 IP를 통해 인터넷에 연결될 수 있도록 NAT(Network Address Translation)를 사용한다.

	짧은 표기법	일반적인 표기법	실제 범위
A class	10.0.0.0/8	10.0.0.0 | 255.0.0.0	10.0.0.0 ~ 10.255.255.255
B class	172.16.0.0/12	172.16.0.0 | 255.240.0.0	172.16.0.0 ~ 172.31.255.255
C class	192.168.0.0/24	192.168.0.0 | 255.255.255.0	192.168.0.0 ~ 192.168.255.255

☞ About IPv6

IPv4에 이은 차세대 IP 주소 표현 방식. **주소 공간을 늘려 망 확장성이 더욱 향상된 IP 주소 체계**로, 휴대폰이나 전자제품에도 적용할 수 있다.
IPv6의 경우, 주소 공간의 확장으로 주소 부족 문제를 해결하였고, **IPSec (Internet Protocol Security) 기능을 자체적으로 제공하여 보안 기능을 향상**시키는 한편 **품질제어와 주소 자동 설정 등의 기능도 추가**하여 관리자와 일반 사용자의 편의성도 증대하였다.

128-bit의 길이로 이루어지는 IPv6는 **2-byte (= 16-bit)씩 8개로 나누어 16진수로 표현**한다.
또한, **나누어진 8개의 16진수는 쌍점(:)으로 구분**한다.
[21DA:00D3:0000:2F3B:02AA:00FF:FE28:9C5A]의 주소에서, "0"을 빼고 단순하게 [21DA:D3:0:2F3B:2AA:FF:FE28:9C5A]로 쓸 수도 있다. 하지만 이 경우, 각 block마다 하나의 숫자는 있어야 하기 때문에 한 block에 "0"만 있을 때 하나는 남겨 두어야 한다.

☞ IPv4와 IPv6의 비교

	IPv4	IPv6
주소 크기	32-bit	128-bit
주소 개수	약 43억	약 43억×43억×43억×43억
주소 표기	8-bit씩 4부분 (10진수 표기) 예) 192.168.0.1	16-bit씩 8부분 (16진수 표기) 예) 2001:230:abcd:ffff:0000:0000:ffff:1111
주소 할당	A, B, C, D 등 class 단위로 구분	네트워크 규모에 따른 순차적 할당
주소 유형	Unicast / Multicast / Broadcast	Unicast / Multicast / Anycast
기본 헤더 필드수	14	8
Header Checksum	있음	기능 삭제
QoS 지원	품질 보장이 곤란	등급별, 서비스별로 packet을 구분할 수 있어 품질 보장용이
보안 기능	IPSec protocol 별로 설치	확장기능에서 기본으로 IPSec 제공
Mobile IP 기능 적용	비효율적	효율적
Plug & Play 기능	없음	있음 (Auto Configuration 가능)

★ QoS (Quality of Service)

- Internet이나 Network 상에서 전송률 및 에러율과 관련된 서비스 품질을 의미.

★ NAT (Network Address Translation)

- 내부 IP 주소를 공인 IP 주소로, 공인 IP 주소를 내부 IP 주소로 바꿔주는 변환 체계.

☞ Subnet mask?

Network Bit와 Host Bit 부분을 구분해 주는 역할을 한다.
사용자가 IP 주소를 사용할 때 지정해 주는 것으로써, 사용자가 임의로 설정하는게 아니라 IP 주소를 할당하는 ISP(Internet Service Provider / 인터넷 서비스 제공업체)에서 제공하는 값을 그대로 입력해야 한다.
왼쪽서부터 bit가 1로 시작하고, 총 32-bit의 길이를 가지며,
Subnet Mask에서 "1"인 부분은 **Network Bit**로, "0"인 부분은 **Host Bit**로 인식한다.

기본적인 Subnet Mask는 Class 별로 다음과 같다.

Class	Subnet
A Class	255.0.0.0
B Class	255.255.0.0
C Class	255.255.255.0

☛ Subnetting?

할당된 기본 IP 주소들을 가지고 내부적으로 여러 개의 Subnet을 만드는 과정.
C Class의 경우, IP 주소의 범위는 [x.x.x.0 ~ x.x.x.255]까지로 총 256개이다.
그러나 "x.x.x.0"은 이 Network를 대표하는 주소로 사용되고, "x.x.x.255"는 이 Network에 대한 Broadcast 주소로 사용되기 때문에 **실제로 사용되는 주소는 254개(256-2)**이다.

기본적인 Subnetting 방법은 "192.168.0.0"으로 설명한다. [C Class]

▶ 192.168.0.0/25
IP 주소 뒤의 "/숫자"는 Network ID의 bit수를 의미한다.
즉, Network ID와 Host ID의 bit 수는 다음과 같다.

192.168.0.0/25 = 11000000.101010000.00000000.00000000
(Subnet Mask) = 11111111.111111111.11111111.10000000 (255.255.255.128)
Network ID 부분 (25-bit) Host ID (7-bit)

Network ID가 1 bit를 빌려서 Host ID 부분의 최상위 1 bit가 "1"로 설정되어 있으므로, 최하위 1 byte는 "128"이라는 10진수를 갖는다. 이 "128"이 Subnetting의 기준 숫자가 된다.
즉, [0 ~ 127] 과 [128 ~ 255] 의 두 부분으로 나눠지게 된다.

첫 번째 Subnet

11000000.101010000.00000000.00000000	=	192.168.0.0	부터 (Network Address)
11000000.101010000.00000000.01111111	=	192.168.0.127	까지 (Broadcast Address)

두 번째 Subnet

11000000.101010000.00000000.10000000	=	192.168.0.128	부터 (Network Address)
11000000.101010000.00000000.11111111	=	192.168.0.255	까지 (Broadcast Address)

하지만 **맨 처음과 맨 끝은 특수 용도로 사용**하기에, 실제로 사용 할 수 있는 IP의 범위는
1 ~ 126, 129 ~ 254 까지 이다.

즉, 새로 생긴 하나의 Subnet 당 실질적으로 Host Computer에 할당할 수 있는 IP 주소의 개수는 [2^7 – 2 = 126] 개가 된다.

★ 여기서 잠깐!!!

▶ 4개의 subnet으로 나눌 경우?

우선, IP 주소의 전체 범위는 0~255까지 "256"개이다.
그 범위를 4개의 subnet으로 분할하면, 한 subnet 당 64개의 IP 주소를 쓸 수 있다.
(256 ÷ 4 = 64)
즉, 0~63, 64~127, 128~191, 192~255 까지 이다.
그런데 앞에서도 이야기 했듯이, 맨 처음과 맨 끝은 특수용도로 쓰이는 부분이므로,
1~62, 65~126, 129~190, 193~254 까지 각 62개씩 할당할 수 있다.

정리하자면, 4개의 subnet으로 나눴을 경우,
각 subnet에 실제 할당 가능한 IP의 갯수는 62개가 된다. (64-2)

▶ 8개의 subnet으로 나눌 경우?

256 ÷ 8 = 32
각 subnet 당 사용 가능한 IP주소의 갯수 : 32
⇒ (0~31, 32~63, 64~95, 96~127, 128~159, 160~191, 192~223, 224~255)
실제 할당 가능한 IP의 갯수 : 30
⇒ (1~30, 33~62, 65~94, 97~126, 129~158, 161~190, 193~222, 225~254)

Subnet	Network Address	Range	Broadcast Address	Netmask
1-Class	x.x.x.0	x.x.x.1 ~ x.x.x.254	x.x.x.255	255.255.255.0 (/24)
2-Class	x.x.x.0 x.x.x.128	x.x.x.1 ~ x.x.x.126 x.x.x.129 ~ x.x.x.254	x.x.x.127 x.x.x.255	255.255.255.128(/25)
4-Class	x.x.x.0 x.x.x.64 x.x.x.128 x.x.x.192	x.x.x.1 ~ x.x.x.62 x.x.x.65 ~ x.x.x.126 x.x.x.129 ~ x.x.x.190 x.x.x.193 ~ x.x.x.254	x.x.x.63 x.x.x.127 x.x.x.191 x.x.x.255	255.255.255.192(/26)
8-Class	x.x.x.0 x.x.x.32 x.x.x.64 x.x.x.96 x.x.x.128 x.x.x.160 x.x.x.192 x.x.x.224	x.x.x.1 ~ x.x.x.30 x.x.x.33 ~ x.x.x.62 x.x.x.65 ~ x.x.x.94 x.x.x.97 ~ x.x.x.126 x.x.x.122 ~ x.x.x.158 x.x.x.161 ~ x.x.x.190 x.x.x.193 ~ x.x.x.222 x.x.x.225 ~ x.x.x.254	x.x.x.31 x.x.x.63 x.x.x.95 x.x.x.127 x.x.x.159 x.x.x.191 x.x.x.223 x.x.x.255	255.255.255.224(/27)

Network Bit	Subnet Mask	Usable IP	Mask / IP / 실사용 IP
25	255.255.255.128(/25)	2^7-2=126	128 / 128 / 실 126
26	255.255.255.192(/26)	2^6-2=62	192 / 64 / 실 62
27	255.255.255.224(/27)	2^5-2=30	224 / 32 / 실 30
28	255.255.255.240(/28)	2^4-2=14	240 / 16 / 실 14
29	255.255.255.248(/29)	2^3-2=6	248 / 8 / 실 6
30	255.255.255.252(/30)	2^2-2=2	252 / 4 / 실 2

☞ OSI Model

[Open Systems Interconnection Reference Model]. 개방형 시스템간 상호접속.
1980년 말, ISO(International Organization for Standardization / 국제 표준화 기구)에 의해 개발된 컴퓨터의 통신절차(Protocol)에 관한 국제 표준규격.

☛ OSI 7-Layer?

통신망을 통한 상호 접속에 필요한 제반 통신 절차를 정의하고 이 가운데 비슷한 기능을 제공하는 module을 7개의 동일 계층으로 분할한 것이다. "OSI 7계층"으로 불리기도 한다.

7 계층	응용 계층	(Application Layer)
6 계층	표현 계층	(Presentation Layer)
5 계층	세션 계층	(Session Layer)
4 계층	전송 계층	(Transport Layer)
3 계층	네트워크 계층	(Network Layer)
2 계층	데이터링크 계층	(Data-link Layer)
1 계층	물리 계층	(Physical Layer)

0x01. 물리 계층 (Physical Layer)
[Data-link Layer] 로부터 전달된 Frame을 0과 1의 bit열로 변환하여 전송 매체를 통해 data를 전송한다. 또한, 사용자 장비와 network 종단 장비 사이의 물리적, 전기적인 interface를 규정하며 전송 선로에 따른 전송 방식과 incoding 방식 등을 결정한다.
간단하게 말해서, 단지 data를 전달할 뿐인 계층이다.
이 계층에 속하는 대표적인 장비로는 [Data cable], [Repeater], [Hub] 가 있다.

0x02. 데이터링크 계층 (Data-link Layer)
[Physical Layer] 를 통해 송/수신되는 정보의 오류와 흐름을 관리하여, 안전한 정보의 전달을 수행 할 수 있도록 도와주는 계층.
[Network Layer] 에서 내려온 packet에 Header와 Trailer를 추가하여 “Frame” 이라는 형태로 만든다.
추가된 Header field에는 data block의 시작을 나타내는 field와 송/수신지의 MAC-address 등이 포함되며, Trailer에는 전송 중 오류 검출을 위한 “오류 검출 코드” 가 삽입된다.
이 계층에 속하는 대표적인 장비로는 [Bridge], [Switch] 가 있다.

0x03. 네트워크 계층 (Network Layer)
data(packet)를 목적지까지 가장 안전하고 빠른 경로와 주소를 정하여 전달해주는 계층.
[Transport Layer]에서 내려온 Segment에 network address 정보를 추가한 후, [Data-link Layer]로 보내주며, 이 data를 “packet” 이라 한다. 또한 data를 송신측에서 수신측까지 안전하게 전송하기 위한 논리적인 link를 설정하고, segment의 크기가 network를 통해 전송할 수 있는 최대 크기인 MTU(Maximum Transmission Unit) 보다 클 경우, 작은 크기의 packet으로 분할하여 전송하는 역할을 수행한다.
이 계층에 속하는 대표적인 장비로는 [Router], [L3 Switch (Routing 기능을 하는 Switch)] 가 있다.

0x04. 전송 계층 (Tranport Layer)
- **전체 메시지의 종단(End-to-End)간 전달, 흐름 제어 및 오류 제어 기능을 수행**하며, 이에 해당하는 대표적인 protocol로는 연결 지향형 서비스인 **[TCP (Transmission Control Protocol)]**와 비연결 지향형 서비스인 **[UDP (User Datagram Protocol)]**가 있다.
- 연결 지향형 (Connection-oriented Service)는 TCP가 대표적인 protocol이며, **송수신 host 사이에 data를 전송하기 이전에 먼저 연결 설정을 맺는 형태**이다. 이러한 절차로 인하여 **신뢰성 있는 data의 전송을 보장**할 수 있다.
- 비연결 지향형 서비스 (Connectionless Service)는 UDP가 대표적인 protocol이며, **data를 전송하기 전에 연결 성정 과정을 거치지 않으므로 빠른 전송 속도를 갖지만, 전송된 data들이 목적지까지 정확하게 전달되었는지에 대한 보장을 받을 수 없게 된다.**

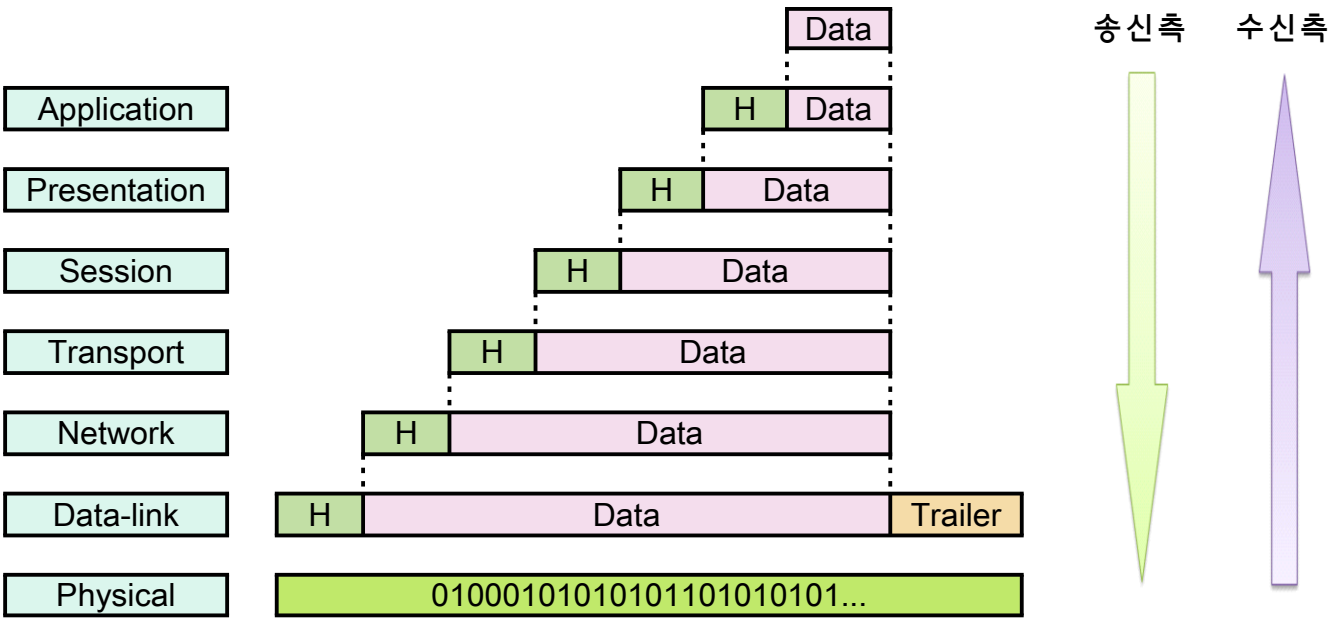
0x05. 세션 계층 (Session Layer)
- **송/수신 host상에서 실행되고 있는 응용 프로그램 간의 session의 확립과 유지, 작업 완료 후 종료 역할을 수행**한다.
- [Session Layer]에서 제공되는 두 system간의 대화 방법으로 **Full-Duplex (전화기)**, **Half-Duplex (무전기)**, **Simplex (라디오)** 등이 있다.

0x06. 표현 계층 (Presentation Layer)
사용자가 입력한 data을 network를 통해 전송 가능한 형태의 data로 변환하는 기능을 담당한다. 예를 들어 문자를 표현할 경우, ASCII code로 할 것인지 EBCDIC code로 할 것인지를 결정하며, **data의 압축과 암호화 기능도 제공**한다.

0x07. 응용 계층 (Application Layer)
사용자가 직접 사용하는 program (e-mail, web browser, file transfer program 등) 들이 동작하는 계층을 말하며, 응용 프로그램이 network에 접근할 수 있는 수단을 제공한다.

☛ Encapsulation / Decapsulation

- 두 개의 host 사이에서 data를 전송할 경우, OSI Model 송신측의 각 계층들은 data를 하위 계층으로 내려보낼 때 Header나 Trailer를 붙여서 내려보낸다. 이렇게 보내진 data는 물리 계층을 통해 상대방 host로 전송된다.
- Data의 수신측에서는 하위 계층으로부터 수신된 Data의 Header와 Trailer를 제거하고 상위 계층으로 올려보내게 되는데, 이와 같이 각 계층에서 Header나 Trailer를 붙이는 과정을 **캡슐화(Encapsulation)**라 하고, 상대방 host에 도착한 Data에서 Header와 Trailer를 제거하는 과정을 **역캡슐화(Decapsulation)**라 한다.



☞ Protocol

서로 다른 개체 간에 의사소통을 하기위해 미리 약속된 규약.

☛ Protocol 의 기능

0x01. 캡슐화 (Encapsulation)
계층 구조에서 상위 계층의 data가 하위 계층으로 전달될 때, 원래의 data에 각종 제어 정보(Header, Trailer 등)들을 추가시키는 것.
추가되는 제어 정보에는 **송/수신자의 주소**, **오류 검출 코드**, **Protocol 제어 정보** 등이 포함된다.

0x02. 오류 제어 (Error Control)
data를 전송하는 과정에서 발생하는 오류를 검출하거나 정정하는 기능. 오류가 발생한 경우나 특정 시간 내에 받지 못한 경우, 재전송을 요구하거나 필요한 조치를 취하게 된다.

0x03. 단편화 (Fragmentation)
network 상에 data를 전송할 경우, 정해진 크기 이상의 data를 일정한 크기의 작은 data block들로 나누어 전송.

0x04. 재조립 (Reassembly)
송신측에서 일정한 크기로 분할하여 전송한 data를 수신측에서 다시 원래의 data로 재조합하는 과정.

0x05. 주소 지정 (Addressing)
data의 정확한 전송을 보장하기 위해 송/수신지의 주소를 부여하는 과정.

0x06. 동기화 (Synchronous)
두 개의 통신하는 개체 사이에 특정 timer 값이나 window size 등의 인자 값을 공유하는 과정.

0x07. 연결 제어 (Connection Control)
두 개의 개체 사이의 data 전송 전, 미리연결 설정 여부에 따라
연결지향형(Connection-oriented) / 비연결형(Connectionless)으로 구분.
- **연결 지향형** : **[연결 설정]**, **[data 전송]**, **[연결 해제]**의 3단계로 구성.
- **비연결형** : **연결 제어 과정 없이** 즉시 data를 전송.

0x08. Multiplexing
하나의 통신회선을 이용하여 여러 통신 개체들이 동시에 data를 전송.

☛ 기본 Protocol - IP

[Internet Protocol]
OSI 참조 모델의 [Network Layer] 에 해당하며, packet 전송을 위한 주소 결정과 이 주소를 사용한 목적지까지의 경로 설정(routing)을 담당.

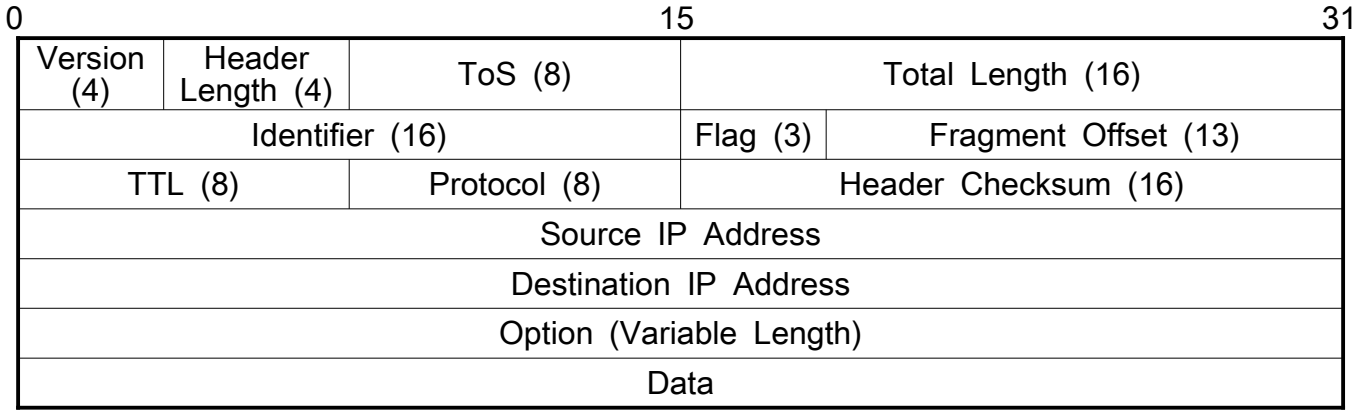
0x01. 특징
◆ **비신뢰성 (Unreliable)**
- 목적지까지의 정확한 packet 전송은 보장하지 않는다. packet 전송의 신뢰성과 관련된 기능들은 TCP에서 제공.

◆ 비접속형 (Connectionless)
- 송/수신 host 사이에 연결 설정 없이 packet을 전송하며, 전송된 packet들은 서로 다른 경로를 경유하여 목적지에 도착할 수도 있다.

◆ 주소 지정
- internet상에 존재하는 host들을 서로 구별할 수 있는 수단으로 IP Address를 사용하며, packet은 IP Address를 기반으로 목적지 host에 전달된다.

◆ 경로 설정
- 목적지 IP Address를 기반으로 어떤 경로를 선택하여 packet을 전송할지를 판단하며, router는 주로 packet의 routing과 관련된 일을 처리한다.

0x02. IP Header 구성



[0000] Version <4-bit>
IP Protocol의 version을 의미한다. 4 or 6.

[0001] Header Length <4-bit>
[Option] field를 포함한 IP Header의 전체 길이를 나타냄.
이 option을 사용하지 않을 경우, 20 bytes 이며, 최대 60 bytes 까지 가능하다.

[0010] Type of Service <8-bit>
data의 우선권을 나타내는 3-bit의 field와 4-bit의 ToS(Type Of Service) field, 1-bit의 예약 field로 이루어진다.
우선권 field는 [0 ~ 7]까지의 숫자로 표시되며, 숫자가 높을수록 더 높은 우선 순위를 갖는다.
TOS field는 최소 지연, 최대 처리량, 최대 신뢰성, 최소 비용을 나타내는 field들로 구성된다.

[0011] Total Length <16-bit>
Header와 data를 포함한 IP packet의 전체 길이를 나타내며, 최대 크기는 65,535 bytes 이다.

[0100] Identifier <16-bit>
전송하고자 하는 data가 network의 최대 전송 크기인 MTU보다 클 경우, packet이 분할되어 전송된다. 이 경우, 분할된 packet은 목적지에서 재조합되는데 동일한 data로부터 분할된 packet들임을 식별하기 위한 식별자 field 값을 갖는다.

[0101] Flag <3-bit>
packet이 분할되었을 때의 제어 정보가 들어가는 field로서, 3-bit로 구성된다.
첫 번째 field는 예약된 field로 사용되지 않고, 두 번째 field는 **단편화 금지** DF(Don't Fragment) bit로, 값이 "1"이면 packet을 분할 할 수 없음을 나타낸다.
만일 packet을 분할하여야 할 경우, 이 값이 "1"이라면 packet을 전송하지 않고 송신측에 ICMP error message를 보낸다.

세 번째의 MF(More Fragment) field는 값이 “1”인 경우, 현재 datagram 다음에 분할된 data 가 추가로 있음을 나타내며, 값이 “0”인 경우는 해당 packet이 분할된 packet 중 마지막 packet임을 나타낸다.

[0110] Fragment Offset <13-bit>
Identifier field와 Flag field를 이용하여 분할된 packet의 수신지에서 packet을 원래의 packet으로 재조립하기 위한, 분할된 packet 간의 순서 정보를 포함.

[0111] TTL (Time-To-Live) <8-bit>
packet이 network상에서 생존할 수 있는 시간을 제한함으로써 network의 불필요한 traffic을 줄이는 역할을 한다.
TTL 값은 packet의 송신측에서 지정하며, packet이 router를 경유할 때마다 값이 1씩 감소된다.
즉, TTL field는 packet이 경유할 수 있는 최대 hop 수를 의미한다.
packet이 전송 도중 값이 0이 되면, packet은 폐기되고 packet의 송신측으로 ICMP error message가 전송된다.
참고로, TTL 값으로 O/S의 종류를 알 수 있다.

TTL	O/S
128	Windows (98, 98 SE / XP / 7 / Server 2000, 2003, 2008)
64	Linux (Ubuntu, RedHat, etc..)
32	Windows (95 / 98)
이 외에도 찾아보면 더 자세히 나온다. 출처1 : http://blog.naver.com/sdream4/10008532246 출처2 : http://libraryz.tistory.com/401	

[1000] Protocol <8-bit>
IP packet이 어떤 상위 protocol을 포함하고 있는지를 나타내는 field.
상위 protocol에는 ICMP, TCP, UDP 등이 있으며, 이 중에서 TCP는 “6”, UDP는 “17”의 값을 갖게 된다.

[1001] Header Checksum <16-bit>
IP packet Header의 오류 발생을 검사하기 위한 field.
data의 빠른 처리를 위해 IP Header에 대해서만 error 검사를 수행한다.

[1010] IP Address (Source / Destination) <32-bit>
32-bit 길이이며, packet의 송/수신측의 IP Address가 기록된다.

[1011] Option
IP Header의 길이는 기본 Header 20-byte에 option으로 40-byte를 더 사용할 수 있다.
시간의 기록, 경로 추적, network 상황 파악 등의 특수한 목적을 위해 사용되며, option field의 길이는 [Header Length] field를 참조하여 알 수 있다.

☛ 기본 Protocol - TCP

[Transmission Control Protocol]
OSI 참조 모델의 [Transport Layer] 에서 동작하며, 송/수신측 사이에 신뢰할 수 있는 data의 전송을 제공.

0x01. 특징

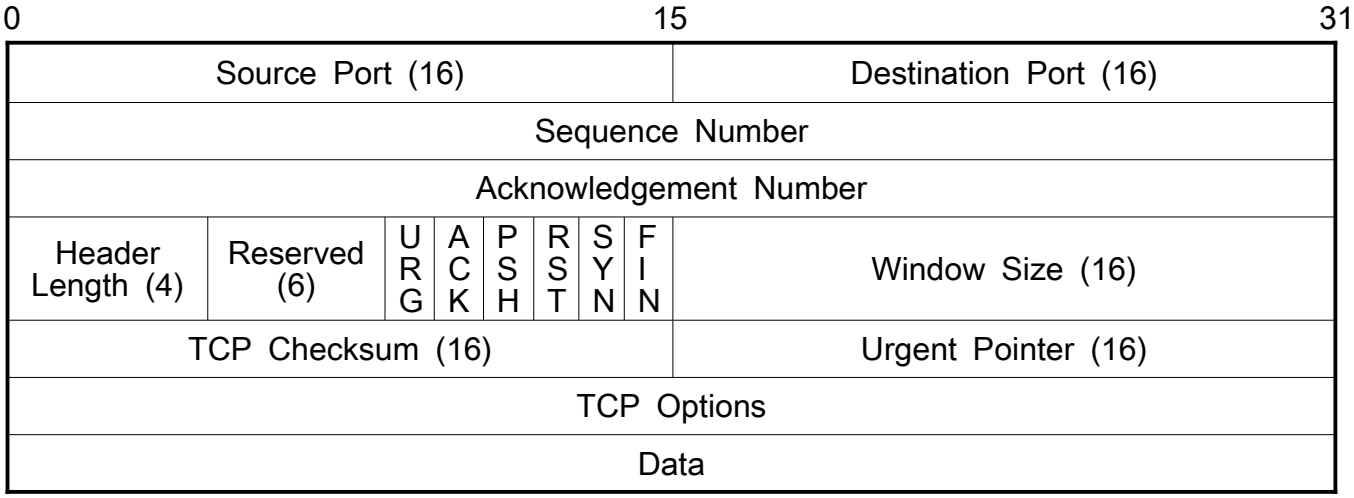
◆ 연결 지향형 (Connection-oriented)

- 송/수신측 사이에 data를 전송하기 전, 연결 설정 과정을 수행. 이를 위해 송신측은 TCP Header의 SYN bit를 “1”로 설정한 packet을 수신측에 전송하여 연결 요청 과정을 진행한다.

◆ 신뢰성 (Reliability)
- 송신측에서 전송한 data가 올바르게 수신측에 도착했는지 **확인 절차를 거쳐 신뢰할 수 있는 data의 전송을 보장.data 전송 후 timer를 동작하여 일정 시간 이내에 송신한 packet에 대한 확인 신호(ACK)가 오지 않을 경우, data를 재전송**한다.

◆ 흐름 제어 (Flow Control)
- 수신측에서 **자신이 처리할 수 있을 만큼의 buffer 공간을 송신측에 알려 이 크기 이상의 data 전송을 제어**함으로써 **buffer overflow를 방지.**

0x02. TCP Header 구성



[0000] Source Port / Destination Port <16-bit>
Segment를 송/수신하는 application의 port number.

[0001] Sequence Number <32-bit>
송신되어지는 data의 순서를 구분하기 위해 사용.

[0010] Acknowledge Number <32-bit>
data의 수신측에서 다음에 전송받고자 하는 byte의 순서 번호.
이 field 값으로 다음 수신할 packet의 순서를 예측할 수 있으며, 수신된 packet의 순서 값에 1을 더한 값을 송신측으로 전송, 수신 확인 기능도 수행한다.

[0011] Header Length <4-bit>
TCP Header의 길이.
4-byte 단위로 Header 길이를 지정하며, 4-bit로 이루어져 **최대 길이는 60 bytes**.

[0100] Reserved <6-bit>
나중에 사용을 위해 예약된 field. "0"으로 설정되어 있음.

[0101] Flag <6-bit>
총 6개의 field로 이루어지며, **TCP packet의 종류를 나타낸다.**

URG	[Urgent Pointer]가 있음.
ACK	[Acknowledge Number]가 유효함.
PSH	가능한 빨리 data를 [Application Layer]로 보내야 함.
RST	연결 재설정.
SYN	연결을 초기화하고, 순서 번호를 동기화.
FIN	송신측에서 data의 전송을 종료.

[0110] Window Size <16-bit>
흐름 제어를 위해 TCP에서 사용하며, 수신측은 자신이 수신 가능한 data의 양을 [Window Size] field에 기술하여 송신측에 알려준다.
송신측에서는 수신측의 ACK를 받지 않아도 maximum window size만큼의 data를 전송할 수 있다.
window size는 16-bit이며, 최대 크기는 65,535 bytes.

[0111] Checksum <16-bit>
TCP Header와 data에 대한 checksum 값이 기록되며, 수신측에서 오류 검출용으로 사용.
TCP Header의 checksum 계산시, TCP Header의 field들에 더해서 송/수신지 IP, protocol, TCP Header size field로 구성된 12-byte의 가상 header를 포함하여 계산한다.

[1000] Urgent Pointer <16-bit>
송신측은 수신측으로 먼저 전송되어야 하는 data가 있을 경우, URG flag를 설정.
사용 예 ⇒ FTP를 이용하여 data를 전송 / {Ctrl+C}를 눌러 전송을 중단하는 경우 등.

[1001] TCP Three-way handshake (TCP 3-way handshake)
data를 전송하기 이전에 송신측과 수신측이 3번에 걸친 message 교환을 통해 연결을 설정하는 과정.
1. Client(송신측)는 Server(수신측)와 연결 성립을 위해 TCP flag의 SYN bit를 "1"로 설정 후, 초기 순서 번호 ISN(Initial Sequence Number)을 설정한 segment를 Server로 전송.
2. Server는 이에 대한 응답으로, SYN bit와 Client가 전송한 ISN에 "1"을 더한 값을 포함한 ACK segment와 자신의 ISN을 전송.
3. 응답 packet을 수신한 Client는 다시 이에 대한 응답으로 ACK를 송신함으로써 Client와 Server 간 연결이 완료된다.

[1010] TCP Four-way handshake (TCP 4-way handshake)
송/수신측 간의 연결을 종료할 때, 4번의 message 교환을 통해 종료.
1. 연결 종료를 원하는 Client(송신측)에서 FIN flag가 설정된 segment를 Server(수신측)에 전송.
2. Server는 이에 대한 확인으로 ACK를 보냄으로써, Clinet에서 Server로의 연결이 종료.
3. Server는 FIN flag가 설정된 segment를 Client로 전송.
4. Clinet는 이에 대한 확인으로 ACK를 보냄으로써, Server에서 Clinet로의 연결이 종료.

☛ 기본 Protocol - UDP

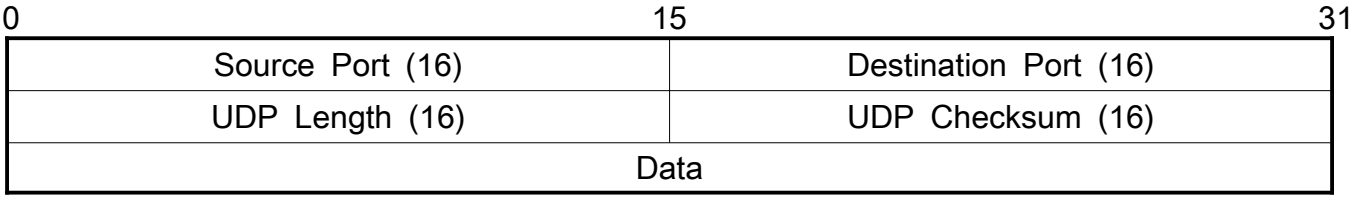
[User Datagram Protocol]
[Transport Layer] protocol로써, 흐름 제어, 수신 확인, 오류 제어 등의 기능이 없고, 제한된 오류 검사와 전송 속도 향상의 특징을 갖는 비연결 지향형(Connectionless) protocol.

0x01. 특징
◆ 비연결형 (Connectionless)
- 연결 설정 과정을 생략,수신측으로 data를 바로 전송.

◆ 최선형 서비스 (Best Effort Service)
- 수신 확인이나 재전송 기능이 없어 packet 손실의 가능성이 있음.

◆ 비상태 정보 (Non-state)
- packet 전송에 대한 연결 정보(순서 번호, 확인 번호, 혼잡 제어, 송/수신 buffer 상태)를 다루지 않으므로, 구성이 간단하고 빠른 동작을 제공.

0x02. UDP Header 구성



[0000] Source Port / Destination Port <16-bit>
Data 송/수신측 process에서 사용하는 port number.

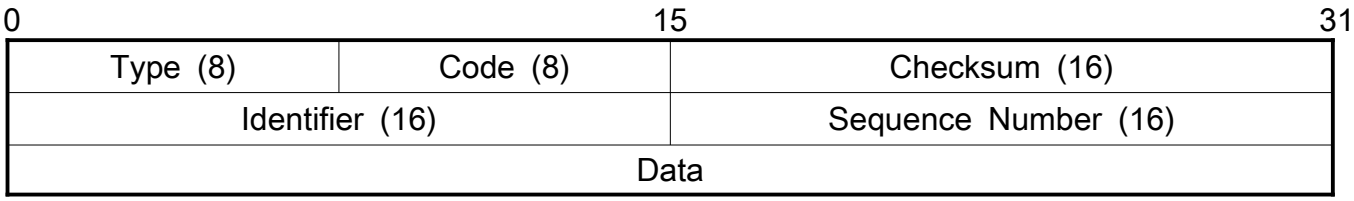
[0001] Message Length <16-bit>
UDP Header와 data를 합친 전체 길이. 최대 65,535 bytes 의 크기를 갖음.

[0010] Checksum <16-bit>
UDP Header와 data를 포함해 오류를 검출하기 위해 사용.

☛ 기본 Protocol - ICMP

[Internet Control Message Protocol]
IP Protocol을 사용하는 network에서 발생할 수 있는 오류에 대한 보고 기능과 network 상태 진단 기능 등을 제공.
[PING(Packet InterNet Groper)], [Tracerouter] 가 ICMP를 사용하는 대표적인 Utility 이다.

0x01. ICMP Header 구성



[0000] Type <8-bit>
ICMP message의 유형을 나타냄.

Type	Message
0	Echo Reply
3	Destination Unreachable
4	Source Quench
5	Redirection Required
8	Echo Request
11	Time Exceeded
12	Parameter Problem
13	Timestamp Request
14	Timestamp Reply
15	Information Request [Not use]
16	Information Reply [Not use]
17	Address Mask Request
18	Address Mask Reply

[0001] Code <8-bit>
message type의 값에 따라 세분화할 필요가 있을 때 사용.

[0010] Checksum <16-bit>
ICMP Header의 오류를 검출하기 위해 사용.

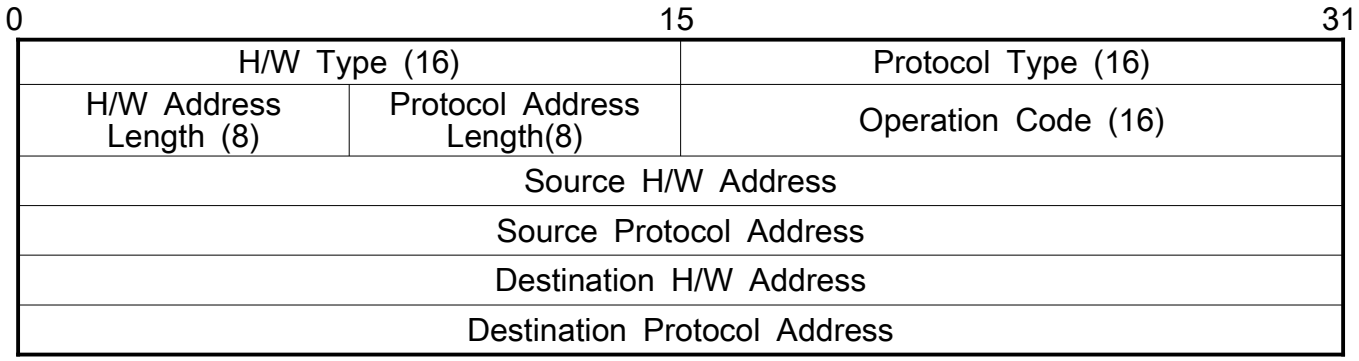
[0011] Identifier <16-bit>
[Code] field에 더 긴 내용이 필요할 경우 사용.

[0100] Data
ICMP message의 유형에 따른 추가적인 정보가 포함.

☛ 기본 Protocol - ARP

[Address Resolution Protocol]
특정 IP Address에 대한 MAC Address 변환 동작을 담당.
반대로, MAC-Address에 대한 IP Address를 얻는 기능을 하는 protocol은 RARP(Reverse ARP)라 한다.

0x01. ARP Header 구성



[0000] H/W Type <16-bit>
ARP를 이용하는 해당 H/W Interface 종류를 정의. Ethernet의 H/W Type은 "1".

[0001] Protocol Type <16-bit>
상위 계층 protocol을 지정. IP에 대한 값은 "0x0800".

[0010] H/W Address Length <8-bit>
물리 주소(MAC-Address)의 길이. 값은 "6".

[0011] Protocol Address Length <8-bit>
논리 주소(IP)의 길이. 값은 "4".

[0100] Operation Code (OP-Code) <16-bit>
ARP 동작을 지정하는 field. 요청은 "1", 응답은 "2"의 값을 갖는다.

[0101] Source H/W / Protocol Address <32-bit>
송신측의 MAC/IP Address

[0110] Destination H/W / Protocol Address <32-bit>
수신측의 MAC/IP Address