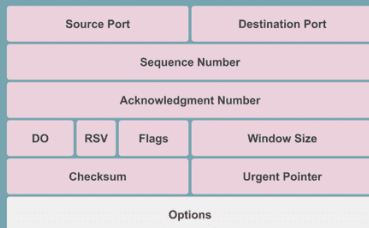


① TCP 헤더

- Transmission Control Protocol Header
- OSI 7계층 중 전송 계층에서 사용되고 있는 프로토콜
- 장비들 간의 통신 과정에서 정보를 안정적으로, 순서대로
에러없이 교환할 수 있도록 하는 것에 목적을 둔 프로토콜

TCP Header																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																											
Offset	Octet	0						1								2								3																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																			
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																										
0	0	Source port																Destination port																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																									
4	32	Sequence number																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																									
8	64	Acknowledgment number (if ACK set)																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																									
12	96	Data offset		Reserved 000			NS	C	E	U	A	P	R	S	F	Window Size																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																											
	W							C	R	C	S	I																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																															
	R							E	G	K	H	T	N	N																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																													



1) Source port, Destination port

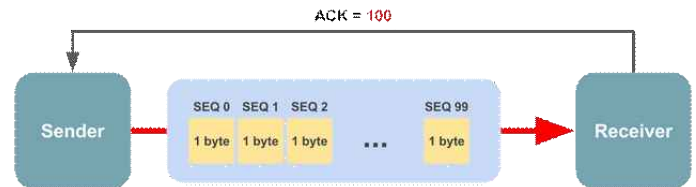
- 세그먼트의 출발지와 목적지를 나타내는 필드
- 각각 16bits를 할당 받음
- 이때 출발지와 목적지의 주소를 판별하기 위해 IP 주소와 포트 번호가 필요
- IP 주소는 당연히 한 계층 밑인 네트워크 계층에 있는 IP의 헤더에 담기기 때문에, TCP 헤더에는 IP 주소를 나타내는 필드가 없고 포트를 나타내는 필드만 존재

2) Sequence Number

- 전송하는 데이터의 순서를 의미
- 32bits를 할당 받음
- 최대 4,294,967,296까지의 수를 담을 수 있음
- 시퀀스 번호가 그리 쉽게 중복되지는 않음
- 수신자는 쪼개진 세그먼트의 순서를 파악하여 올바른 순서로 데이터를 재조립할 수 있게 됨
- 송신자가 최초로 데이터를 전송할 때는 이 번호를 랜덤한 수로 초기화
- 이후 자신이 보낼 데이터의 1bytes당 시퀀스 번호를 1씩 증가시키며 데이터의 순서를 표현
- 4,294,967,296을 넘어갈 경우 다시 1부터 재시작

3) Acknowledgment Number

- 승인 번호는 데이터를 받은 수신자가 예상하는 다음 시퀀스 번호를 의미
- 32bits를 할당 받음
- 연결 설정과 연결 해제 때 발생하는 핸드셰이크 과정에서는 상대방이 보낸 시퀀스 번호 + 1로 자신의 승인 번호를 만들어내지만, 실제로 데이터를 주고 받을 때는 상대방이 보낸 시퀀스 번호 + 자신이 받은 데이터의 bytes로 승인 번호를 만들어 냄
- 1MB짜리 같은 큰 데이터를 한 번에 전송할 수는 없으므로 송신자는 이 데이터를 여러 개의 세그먼트로 쪼개서 조금씩 전송해야 함
- 이때 송신자가 한 번에 전송할 수 있는 데이터 양은 네트워크나 수신자의 상태에 따라 가변적임
- 시퀀스 번호는 1bytes당 1씩 증가하기 때문에 첫 번째 바이트 문치는 0, 두 번째 바이트 문치는 1, 세 번째 바이트 문치는 2와 같은 순서로 매겨질 것이다.



4) Data Offset

- 전체 세그먼트 중에서 헤더가 아닌 데이터가 시작되는 위치가 어디부터인지를 표시
- 이 오프셋을 표기할 때는 32비트 워드 단위를 사용
- 32비트 체계에서의 1word = 4bytes를 의미
- 이 필드의 값에 4를 곱하면 세그먼트에서 헤더를 제외한 실제 데이터의 시작 위치를 알 수 있음
- 이 필드에 해당된 4bits로 표현할 수 있는 값의 범위는 0000 ~ 1111, 즉 0 ~ 15word 이므로 기본적으로 0 ~ 60bytes의 오프셋까지 표현할 수 있음
- 옵션 필드를 제외한 나머지 필드는 필수로 존재해야 하기 때문에 최소 값은 20bytes, 즉 5word로 고정되어 있음
- 이 필드는 옵션(Optional) 필드의 길이가 고정되어 있지 않기 때문에 필요

5) Reserved (3bits)

- 미래를 위해 예약된 필드
- 모두 0으로 채워져야 함

6) Flags (NS ~ FIN)

- 9개의 비트 플래그
- 현재 세그먼트의 속성을 나타냄
- 기존에는 6개의 플래그만을 사용했지만 혼잡 제어 기능의 향상을 위해 Reserved 필드를 사용하여 NS, CWR, ECE 플래그가 추가되었음.

* 필드의 의미

필드	의미
URG	<ul style="list-style-type: none"> - Urgent Pointer(긴급 포인터) 필드에 값이 채워져있음을 알리는 플래그 - 이 포인터가 가리키는 긴급한 데이터는 높게 처리되어 먼저 처리된다. - 요즘에는 많이 사용되지 않음
ACK	<ul style="list-style-type: none"> - Acknowledgment(승인 번호) 필드에 값이 채워져있음을 알리는 플래그 - 이 플래그가 0이라면 승인 번호 필드 자체가 무시된다.
PSH	<ul style="list-style-type: none"> - Push 플래그 - 수신 측에게 이 데이터를 최대한 빠르게 응용프로그램에게 전달해달라는 플래그 - 이 플래그가 0이라면 수신 측은 자신의 버퍼가 다 채워질 때까지 기다림 - 이 플래그가 1이라면 이 세그먼트 이후에 더 이상 연결된 세그먼트가 없음을 의미하기도 함
RST	<ul style="list-style-type: none"> - Reset 플래그 - 이미 연결이 확립되어 ESTABLISHED 상태인 상대방에게 연결을 강제로 리셋해달라는 요청의 의미
SYN	<ul style="list-style-type: none"> - Synchronize 플래그 - 상대방과 연결을 생성할 때, 시퀀스 번호의 동기화를 맞추기 위한 세그먼트임을 의미한다.
FIN	<ul style="list-style-type: none"> - Finish 플래그 - 상대방과 연결을 종료하고 싶다는 요청인 세그먼트임을 의미한다.

※ 기존의 Reserved 필드를 사용하여 새롭게 추가된 NS, CWR, ECE 플래그는 네트워크의 명시적 혼잡통보(Explicit Congestion Notification, ECN)을 위한 플래그

※ ECN을 사용하지 않던 기존의 네트워크 혼잡 상황 인지 방법은 타임아웃을 이용한 방법이었음

7) Windows Size

- 한번에 전송할 수 있는 데이터의 양을 의미하는 값을 담음
- 2^{16} = 65535만큼의 값을 표현할 수 있음
- 단위는 바이트
- 윈도우의 최대 크기는 64KB라는 말이 echo

8) Checksum

- 데이터를 송신하는 중에 발생할 수 있는 오류를 검출하기 위한 값
- TCP의 체크섬은 전송할 데이터를 16Bits씩 나눠서 차례대로 더해가는 방법으로 생성
- 방식은 단순

9) Urgent Pointer

- 긴급 포인터
- URG 플래그가 1이라면 수신 측은 이 포인터가 가르키고 있는 데이터를 우선 처리

필드	의미
NS	- ECN에서 사용하는 CWR, ECE 필드가 실수나 악의적으로 은폐되는 경우를 방어하기 위해 RFC 3540에서 추가된 필드
ECE	<ul style="list-style-type: none"> - ECN Echo 플래그 - 해당 필드가 1이면서, SYN 플래그가 1일 때는 ECN을 사용한다고 상대방에게 알리는 의미 - SYN 플래그가 0이라면 네트워크가 혼잡하니 세그먼트 윈도우의 크기를 줄여달라는 요청의 의미
CWR	- 이미 ECE 플래그를 받아서, 전송하는 세그먼트 윈도우의 크기를 줄였다는 의미

10) Options

- TCP의 기능을 확장할 때 사용하는 필드들
- 크기가 고정된 것이 아니라 가변적
- 수신 측이 어디까지가 헤더고 어디서부터 데이터인지 알기 위해 위에서 설명한 데이터 오프셋 필드를 사용

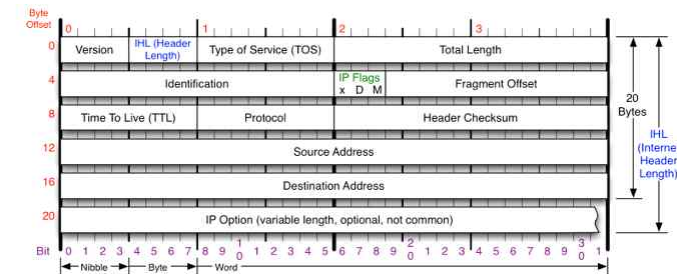
② IP 헤더

- IP Header
- Layer 3 계층 프로토콜
- IP는 로컬 환경에서 리모트 환경으로 데이터 전송을 담당
- 경로를 선출하여 선출된 최적의 경로를 이용하여 데이터 전송 담당
- 대신 UDP와 마찬가지로 비연결 지향성 프로토콜(Connectionless Service)이며 Best Effort Service를 지향하기 때문에 신뢰성은 제공하지 않음

IP Header 20Byte				Data
① Version (4bit)	② Head Length (4bit)	③ Type of Service (8bit)	④ Total Length (16bit)	
⑤ Identification (16bit)	⑥ Flag (3bit)	⑦ Fragmentation offset (13bit)		
⑧ Time to Live (8bit)	⑨ Protocol (8bit)	⑩ Header Checksum (16bit)		
⑪ Source IP Address (32bit)				
⑫ Destination IP Address (32bit)				
⑬ Option				

- ① IP 프로토콜 버전을 표기하며 이 때 버전은 IP Version 4와 IP Version 6가 존재한다.
- ② IP 헤더의 길이를 표기하며 20Byte에서 최대 60Byte이다.
- ③ QOS (서비스 품질)에 대한 정보를 제공한다. (데이터 우선순위 확인)
- ④ IP 데이터그램 전체 길이를 Byte로 표기한다. (데이터 길이 = 전체 길이 - 헤더 길이)
- ⑤ 호스트들 간에 전송하는 데이터를 표기한다. 데이터그램을 식별하는 역할을 한다.
- ⑥ IP 데이터그램 단편화에 대한 정보를 갖고 있다. 즉 IP 데이터그램이 분할될 수 있는지를 판단하고 만약 분할된 경우 또 다른 분할된 IP 데이터그램이 있는지 검사한다.
- ⑦ 단편화된 데이터그램들을 하나의 데이터그램으로 합칠 때 전체 데이터그램에서의 위치를 표기한다.
- ⑧ IP 데이터그램이 지날 수 있는 라우터 개수에 대한 생존 시간을 기록한다. (Cisco=255, Windows=128, Linux=64이며 라우터를 지날때 마다 1씩 감소한다.)
- ⑨ IP 데이터그램의 사용자 데이터에 표기된 상위 계층 프로토콜을 의미한다.
- ⑩ IP 헤더 손상 여부를 판단한다.
- ⑪, ⑫ 출발지, 목적지 IP 주소 필드이다.
- ⑬ 옵션 필드이다.

③ IPv4 헤더



Version	Protocol	Fragment Offset	IP Flags
Version of IP Protocol. 4 and 6 are valid. This diagram represents version 4 structure only.	IP Protocol ID. Including (but not limited to): 1 ICMP 17 UDP 57 SKIP 2 IGMP 47 GRE 88 EIGRP 6 TCP 50 ESP 89 OSPF 9 IGMP 51 AH 115 L2TP	Fragment offset from start of IP datagram. Measured in 8 byte (2 words, 64 bits) increments. If IP datagram is fragmented, fragment size (Total Length) must be a multiple of 8 bytes.	x D M x 0x80 reserved (evil bit) D 0x40 Do Not Fragment M 0x20 More Fragments follow
Header Length	Total Length	Header Checksum	RFC 791
Number of 32-bit words in TCP header, minimum value of 5. Multiply by 4 to get byte count.	Total length of IP datagram, or IP fragment if fragmented. Measured in Bytes.	Checksum of entire IP header	Please refer to RFC 791 for the complete Internet Protocol (IP) Specification.

1) Version 필드 (4bit)

- TCP/IP 제품은 IPv4를 사용

2) Header Length 필드 (4bit)

- IP 헤더의 길이를 32비트 단위로 나타낸다.
- 대부분의 IP 헤더의 길이는 20바이트
- 필드 값은 거의 항상 5
- (5 * 32 = 160bit or 20Byte)

3) Type-of-Service Flags

- 서비스의 우선 순위를 제공

값	서비스	설명
0	보통(normal)	보통 데이터로 취급. 대부분의 IP 데이터는 이 분류에 속한다.
1	비용최소화 (Minimize Cost)	IP가 가장 비용이 적게 드는 경로를 통해 패킷을 라우트할 것을 요구하는 경우 사용. 플래그 0 - 보통 1 - 저비용
2	신뢰성 최대화(Maximize Reliability)	IP가 가장 신뢰성 있는 네트워크를 통해 패킷을 라우트할 것을 요구하는 경우 사용. 플래그 0 - 보통 1 - 높은신뢰성
4	처리량 최대화(Maximize Throughput)	IP가 보통 데이터 패킷을 더 많은 처리를 하면서 라우트할 것을 요구하는 경우 사용. 플래그 0 - 보통 1 - 높은 처리량
8	지연 최소화(Minimize Delay)	IP가 보통 데이터 보다 더 신속한 패킷의 라우트를 요구 하는 경우 사용 자신의 입력 에코(echo)를 신속하게 보고 싶어하는 Telnet과 같은 애플리케이션 등에 유용 플래그 0 - 보통 1 - 지연 줄임
15	보안 최대화(Maximize Security)	IP가 가장 안전한 경로를 통해 패킷을 라우트할 것을 요구할 수 있다고 한다. 실험용이므로 이 설정을 지원하지 않는 공급 업체가 많다.

4) Total Packet Length 필드 (16bit)

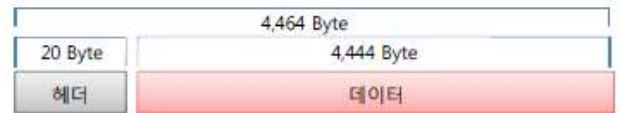
- 전체 IP 패킷의 길이를 바이트 단위로 나타낸다.

5) Fragmentation Flags 필드 (3bit)

- 처음 1bit는 항상 0으로 설정
- May Fragment : IP 라우터에 의해 분열되는 여부를 나타냄
- 플래그 : 0 / 분열 가능 : 1 / 분열 방지
- More Fragments : 원래 데이터의 분열된 조각이 더 있는지 여부 판단
- 플래그 : 0 - 마지막 조각 / 기본값 1 - 조각이 더 있음

6) Fragmentation Offset 필드 (13bit)

- 8바이트 오프셋으로 조각에 저장된 원래 데이터의 바이트 범위



7) Time-to-live 필드 (8bit)

- 데이터를 전달할 수 없는 것으로 판단되어 소멸되기 이전에 데이터가 이동할 수 있는 단계의 수를 나타낸다.
- Time-to-live 필드는 1에서 255사이의 값을 지정하며 라우터들을 패킷을 전달할 때마다 이 값을 하나씩 감소시킨다.

8) Protocol Identifier 필드 (8bit)

- 상위 계층 프로토콜
- 1 - ICMP, 2 - IGMP, 6 - TCP, 17 - UDP

9) Header Checksum 필드 (16bit)

- IP 헤더의 체크섬을 저장, 라우터를 지나갈 때마다 재계산을 하기 때문에 속도가 떨어짐

10) Source IP Address 필드 (32bit)

- 출발지 IP 주소

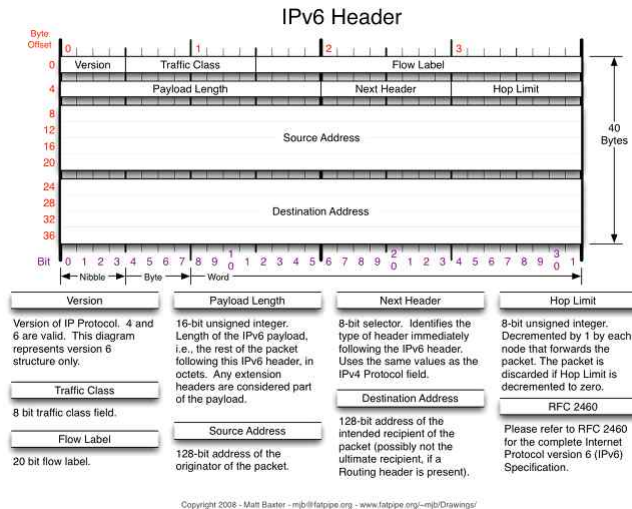
11) Destination IP Address 필드 (32bit)

- 목적지 IP 주소

12) Options(선택적) 필드(가변적)

- Type-of-service 플래그처럼 특별한 처리 옵션을 추가로 정의할 수 있다.

④ IPv6 헤더



1) IPv6 기본 헤더

- 확장 헤더를 포함하지 않은 경우의 기본 헤더(40 바이트)

2) IPv6 확장 헤더

- 기본 고정 헤더 뒤 페이로드 내에 선택적인 확장 헤더들이 뒤따라옴

3) IPv6 기본 헤더 필드 (8개)

- Version (4 비트)
 - IPv4 이면 4 (0100), IPv6 이면 6 (0110) [참고_웹] [IANA 버전 관리](#)
- Traffic Class 또는 Priority (8 비트)
 - IPv4일때의 TOS 필드와 유사
 - . IP 패킷 마다 서로다른 서비스 요구사항을 구분하기 위함
 - . 따라서, 민감한 실시간 응용 및 긴급하지 않은 데이터 패킷 간의 차별적 구분 가능
- Flow Label (20 비트)
 - IP를 연결지향적 프로토콜로 사용할 수 있게 함
 - . 실시간 서비스 등 같이 우선권을 주기위하여 특정 트래픽 Flow에 대한 라벨링
- Payload length (16 비트)
 - 페이로드의 길이 (확장헤더 + 상위계층 데이터) < 2¹⁶(65536) 까지 가능
- Next header (8 비트) [IPv6 확장헤더 참조](#)
 - 기본헤더 다음에 위치하는 확장 헤더의 종류를 표시
 - . IPv4의 프로토콜 번호와 같은 역할
 - .. 예) 0 : Hop-by-Hop Options Header for IPv6
 - .. 예) 58 : ICMPv6 (ICMP for IPv6)
- Hop limit (8 비트)
 - 버전 4일때의 TTL과 같은 역할

○ Source address (128 비트)

- 발신처 주소

○ Destination address (128 비트)

- 목적지 주소

. 만일, 소스 라우팅일 경우에 다음 라우터 주소를 나타냄