IT CookBook, 쉽게 배우는 데이터 통신과 컴퓨터 네트워크(개정) 8장 연습문제 해답

1. 128
2. Hop Limit
3. Flow Label
4. 캡슐
5. ① COA, ② 터널
6. ① ARP, ② RARP
7. ICMP, 송신
8. 같은, IP 패킷
9. ① Checksum, ② Group Address

10. ②, ③, ④, ⑤

(설명①) 송신 호스트와 수신 호스트 주소를 표시하는 공간의 크기가 32 비트에서 128 비트로 확장되었다.

11. ①, ③, ④

(설명②) Fragment Header 확장 헤더는 IPv4 프로토콜의 Fragment Offset, Identification, MF 필드처럼 패킷 분할과 관련된 정보를 포함한다.

(설명⑤) Hop Limit 필드는 IPv4의 Time To Live 필드와 동일한 역할을 수행한다. 즉, 이 값은 패킷이 라우터에 의해 중개될 때마다 감소되며, 0이 되면 해당 패킷은 네트워크에서 사라진다.

12. ①, ②, ④

(설명③) Link/Site 지역 주소 공간은 지역적으로 사용하는 주소로, 개별 지역에서 사용하므로 외부와 충돌이 발생하지 않는다.

(설명⑤) 멀티캐스트 주소 공간에서는 128 비트의 전체 주소 필드가 상위 8 비트의 1111 1111과 4 비트의 플래그, 4 비트의 스코우프 플래그, 112 비트의 그룹 구분자로 구성된다.

13. ②

(설명②) 이동 호스트에는 고유 IP 주소인 홈 주소가 할당되고, 이 주소는 호스트 위치가 바뀌어도 변하지 않는다.

14. ②, ⑤

(설명②) 사용자는 일반적으로 도메인 이름을 입력하는데, 도메인 이름은 DNS 서비스를 통해 IP 주소로 쉽게 변환할 수 있다.

(설명⑤) RARP 프로토콜은 MAC 주소를 이용해 IP 주소를 변환해주는 기능을 수행한다.

15. ①, ②, ③, ⑤

(설명④) TIME EXCEEDED 메시지는 패킷의 TTL 필드 값이 0이 되어 패킷이 버려지는 경우에 주로 발생한다.

16. ③

(설명③) 다중 수신 호스트를 표시하는 멀티캐스트 그룹 주소 표기 방법을 통일해야 한다. IPv4와 IPv6 프로토콜은 이 형식의 주소를 지원한다.

17. ②, ⑤

(설명②) IGMP 메시지는 ICMP 메시지처럼 IP 패킷에 실려서 전달된다.

(설명⑤) 개별 호스트가 자신의 그룹 멤버를 유지하려면 IGMP 보고 메시지를 사용해 IGMP 질의에 응답해야 한다. 만일 호스트의 응답이 이루어지지 않으면 해당 호스트는 그룹에서 탈퇴한 것으로 간주한다.

18.

IPv4와 비교하였을 때, IPv6 프로토콜의 주요 기능은 다음과 같다.

■ 주소 공간 확장 송신 호스트와 수신 호스트의 호스트 주소를 표시하는 공간이 32비트에서 128비트로 확장되었다. IPv6를 이용한 인터넷 환경에서는 이론적으로 호스트를 최대 2128개까지 지원하여 폭발적으로 증가하는 인터넷 사용자를 수용할 수 있다. 또한 개인이 무선으로 사용하는 유비쿼터스(Ubiquitous) 장비가 기하급수적으로 보급되는 환경에도 쉽게 대처할 수 있다.

■ 헤더 구조 단순화

IPv4의 헤더 구조는 매우 복잡하다. 반면, IPv6 헤더에서는 현대의 네트워크 환경에서 불필요한 필드가 제외되거나 확장 헤더 형식으로 변경되었다. 이는 기존의 IP 프로토콜에서 과도하게 수행하는 오류 제어 같은 오버헤드를 줄여 프로토콜의 전송 효율을 높이기 위함이다.

■ 흐름 제어 기능 지원

흐름 제어 기능을 지원할 수 있는 필드(Flow Label 등)를 도입해 일정 범위 내에서 예측 가능한 데이터 흐름을 지원한다. 따라서 하나의 연속 스트림(Stream)으로 전송해야 하는 연관 패킷의 전송 기능을 지원함으로써 실시간 기능이 필요한 멀티미디어 응용 환경을 수용할 수 있다.

19.

IPv6 프로토콜의 헤더 구조는 IPv4보다 매우 단순해 기본 필드를 8개 지원한다. 그림처럼 총 40바이트 중 32바이트는 주소 공간으로 할당하고, 8바이트만 프로토콜의 기능을 위해 사용한다. IPv6 프로토콜의 패킷 헤더는 기본 헤더와 확장 헤더로 나누어지는 데, 그림은 크기가 고정된 기본 헤더의 구조며, 상단의 숫자는 크기를 나타내는 비트 수다.

그림 8-1 IPv6 기본 헤더의 구조

■ 확장 헤더

IPv6 기본 헤더 바로 뒤에 확장 헤더를 하나 이상 둘 수 있는데, 확장 헤더의 종류는 다

음과 같다.

- Hop-by-Hop Options Header : Jumbo 페이로드 옵션과 라우터 긴급 옵션 등과 같은 hop-by-hop 옵션의 처리를 지원한다. Jumbo 페이로드 옵션은 패킷 데이터의 크기가 65,535바이트보다 클 때 사용하는데, 라우터에서 이 패킷을 처리할 수 없으면 ICMPv6 프로토콜의 오류 메시지가 발생한다. 라우터 긴급 옵션은 라우터에 전송 대역 예약 같은 특정 정보를 제공한다.
- Routing Header : IPv4의 소스 라우팅과 유사한 기능을 제공하는데, 패킷이 Routing Header에 지정된 특정 노드를 경유하여 전송되도록 한다. 즉, 헤더에 있는 주소 목록의 첫 번째 호스트에 패킷이 전송되면, 패킷을 받은 호스트가 헤더 목록을 다시 검사해 두 번째 호스트에 전송한다. 이 과정을 패킷이 최종 목적지에 도착할 때까지 반복한다.
- Fragment Header : IPv4 프로토콜 헤더에 정의된 Fragment Offset, Identification, MF 필드처럼 패킷 분할과 관련된 정보를 포함한다.
- Destination Options Header : 수신 호스트가 확인할 수 있는 옵션 정보를 제공한다.
- Authentication Header : 패킷 인증 관련 기능을 제공한다.
- Encapsulating Security Payload Header : 프라이버시 기능을 제공하기 위해 페이로드를 암호화한다. 인증된 목적지 호스트에서 암호화 데이터를 해독할 수 있는 정보도 함께 제공한다.

■ Priority 필드

Priority 필드는 송신 호스트가 패킷을 전송할 때 특정 패킷의 우선순위를 높이는 용도로 사용한다. 우선순위는 동일 송신 호스트가 전송한 패킷에 상대적으로 적용된다. 혼잡 제어 유무에 따라 송신 호스트가 전송하는 패킷은 Priority 필드를 이용해 두 가지로 구분하여 처리할 수 있다.

■ Flow Label 필드

Flow Label 필드는 음성이나 영상 데이터처럼 실시간 서비스가 필요한 응용 환경에서 사용하는데, 기본 원칙은 다음과 같다.

- Flow Label 필드를 지원하지 않는 호스트나 라우터에서는 IPv6 패킷을 생성할 때 반드시 0으로 지정해야 한다. 패킷의 중개 과정에서는 현재 값을 그대로 유지하며, 패킷을 수신하는 측에서는 필드 값을 무시한다.
- Flow Label 필드의 값이 0 이외의 동일한 번호로 부여받은 패킷은 Destination Address, Source Address, Priority, Hop-by-Hop Options Header, Routing Header 등을 모두 동일하게 지정해야 한다. 패킷을 중개하는 라우터가 다른 필드 값을 보지 않고 Flow Label 필드만으로 라우팅 등을 간단히 처리하도록 하기 위함이다.
- Flow Label 필드 값은 1~224-1에서 랜덤하게 선택된다. 단, 현재의 전송 흐름에서는 동일 번호가 부여되지 않도록 해야 한다.

■ 기타 필드

IPv6 프로토콜의 기본 헤더에 정의된 나머지 필드의 의미는 다음과 같다.

- Version Number(버전 번호) : IP 프로토콜의 버전 번호다. 기존 IPv4와 구분하기 위해 6으로 지정된다.
- Payload Length(페이로드 길이) : 헤더를 제외한 패킷의 크기로 단위는 바이트다.
- Next Header(다음 헤더) : 기본 헤더 다음에 이어지는 헤더의 유형을 수신 호스트에게 알려 준다. Next Header에 표시할 수 있는 헤더는 IPv6의 확장 헤더일 수도 있고, 상위 계층인 TCP와 UDP의 헤더일 수도 있다. TCP와 UDP 헤더가 위치하면 확장 헤더가 사용되지 않은 경우다.
- Hop Limit(홉 제한) : IPv4 프로토콜의 Time To Live 필드와 동일한 역할을 수행한다. 이 값은 패킷이 라우터에 의해 중개될 때마다 감소되며, 0이 되면 해당 패킷은 네트워크에서 사라진다.
- Source Address/Destination Address(송신 호스트 주소/수신 호스트 주소) : 송수신 호스트의 IP 주소를 나타낸다. 최대 128비트를 지원한다.

20.

IPv6 프로토콜에서는 특정 송수신 호스트 사이에 전송되는 데이터를 하나의 흐름으로 정의해 중간 라우터가 이 패킷을 특별한 기준으로 처리할 수 있도록 지원한다. 따라서 라우터는 이 기능을 지원하기 위해 필요한 흐름 정보를 저장하여 처리할 수 있어야 한다.

Flow Label 필드는 음성이나 영상 데이터처럼 실시간 서비스가 필요한 응용 환경에서 사용하는데, 기본 원칙은 다음과 같다.

- Flow Label 필드를 지원하지 않는 호스트나 라우터에서는 IPv6 패킷을 생성할 때 반드시 0으로 지정해야 한다. 패킷의 중개 과정에서는 현재 값을 그대로 유지하며, 패킷을 수신하는 측에서는 필드 값을 무시한다.
- Flow Label 필드의 값이 0 이외의 동일한 번호로 부여받은 패킷은 Destination Address, Source Address, Priority, Hop-by-Hop Options Header, Routing Header 등을 모두 동일하게 지정해야 한다. 패킷을 중개하는 라우터가 다른 필드 값을 보지 않고 Flow Label 필드만으로 라우팅 등을 간단히 처리하도록 하기 위함이다.
- Flow Label 필드 값은 $1 \sim 2^{24}-1$에서 랜덤하게 선택된다. 단, 현재의 전송 흐름에서는 동일 번호가 부여되지 않도록 해야 한다.

21.

IPv6 프로토콜에서 지원하는 128비트의 숫자는 아주 커 [그림 8-2]와 같이 16비트의 숫자 8개를 콜론(:)으로 구분한다.

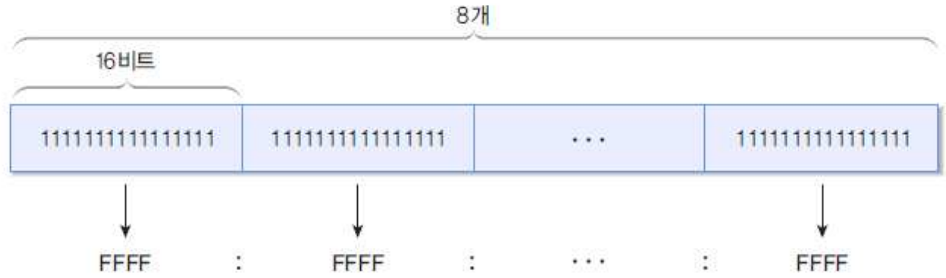


그림 8-2 IPv6의 주소 표현

예를 들어, D1D1:1111:3F3F:1700:4545:1212:1111:1231처럼 표현할 수 있다. 이처럼 IPv6 주소(IPv6 Address)는 아주 커 일일이 표기하기 불편하므로 축약해 표시하는 방안도 만들어지고 있다. 아울러 IPv4 프로토콜과 함께 사용하는 환경에서 IPv4 주소를 캡슐화하여 다음과 같이 표현하기도 한다. X:X:X:X:X:X에서 X는 16비트므로 총 96(16×6)비트고, d.d.d.d에서 d는 8비트므로 총 32(8×4)비트다. 따라서 전체 크기는 IPv6의 주소 크기와 동일한 128(96+32)비트다.

X:X:X:X:X:X:d.d.d.d

22.

터널링 원리를 이해하기 위해 육지와 섬을 거쳐 사람이 이동하는 경우를 예로 들어보자. 홍길동이라는 사람이 육지 a 지점에서 출발하여 섬에 위치한 목적지 d에 도착하려면 중간의 b와 c를 거쳐야 한다. 이동 시 육지에서는 버스를 이용하고, 바다에서는 배를 이용한다. 이때 버스와 배는 모두 본체을 지원하는 IP 프로토콜이라고 볼 수 있으며, 버스와 배에 실려서 이동하는 홍길동은 IP 프로토콜의 전송 데이터가 된다([그림_상이한 전송 수단]).

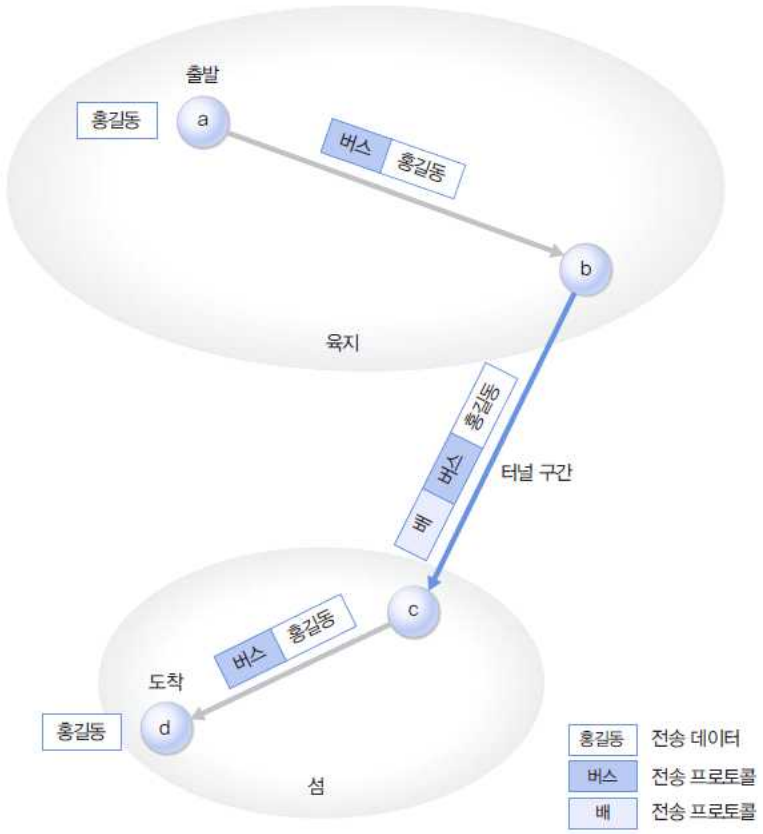


그림 8-4 IP 터널링의 원리

이 방식의 문제점은 출발지 a에서 목적지 d에 도착하는 과정에서 홍길동은 버스라는 IP 프로토콜에서 배라는 IP 프로토콜로 갈아타야 한다는 것이다. 즉, 홍길동 스스로 IP 프로토콜을 교체하는 작업이 추가로 이루어져야 한다.

터널링 방식

IP 프로토콜 교체보다 문제를 간단히 해결하는 방법은 [그림_IP 터널링 원리]와 같은 터널링 기능을 이용하는 것이다. 터널링 방식을 사용하면 홍길동이 출발지에서 목적지까지 버스만 이용하므로 네트워크 최종 사용자인 홍길동이 IP 프로토콜의 교체 과정에 개입하지 않는다는 장점이 있다. 중간의 바다에서는 버스 회사가 배를 직접 처리하여 버스가 배를 타는 형태의 터널 기능을 지원한다. 결과적으로 홍길동은 출발지에서 버스를 타고, 도착지에서는 버스에서 내리기만 하면 된다. 홍길동이 버스를 타고 내리는 중간 내내 잠을 자는 경우를 생각해보면 배라는 프로토콜을 전혀 이해하지 못해도 문제가 되지 않는다.

터널링 방식은 홍길동의 출발과 도착이 이루어지는 a와 d지점의 처리 방식이 [그림_상이

한 전송 수단]과 정확히 일치하며, 터널링이 필요한 지점의 추가 작업은 홍길동이 아닌 제삼자가 처리하는 구조다. 이동 IP를 처리하는 과정에서 사용자는 터널링 관련 부분에 대한 부담을 갖지 않는다. [림_상이한 전송 수단]]과 [그림_IP 터널링 원리에서 사용하는 프로토콜인 배와 버스는 경로 문제를 처리하는 수단으로, 모두 IP 프로토콜로 간주할 수 있다.

23.

홈 에이전트와 이동 에이전트 사이에 설정된 터널(Tunnel)은 그림과 같이 원 IP 패킷을 목적지까지 전송하기 위한 중간 단계의 새 경로다. 따라서 송신 호스트와 수신 호스트 사이에서 동작하는 IP 프로토콜과는 별도의 프로토콜을 사용해 패킷을 중개해야 한다.

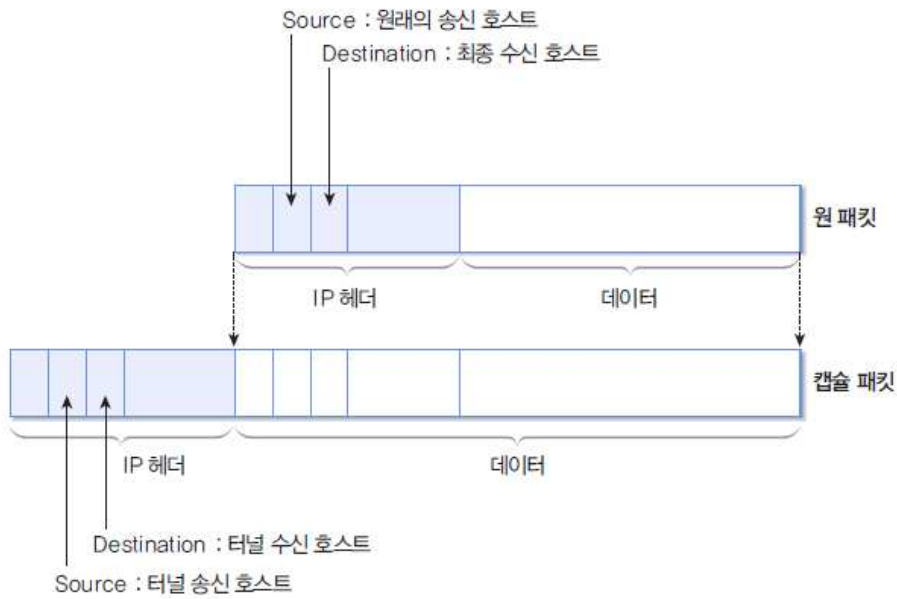


그림 8-6 IP 터널

터널 구간을 지나는 과정에서 라우팅 처리가 필요한데, 여기서는 IP 프로토콜을 사용해야 한다. 원 IP 패킷을 데이터로 취급하는 새로운 형태의 IP 캡슐 패킷(IP Capsule Packet)이 구성되어 전달된다. 원 패킷의 Destination Address 필드에는 이동 호스트의 홈 주소가 들어간다. 홈 에이전트에서는 원 패킷을 이동 호스트에 전달하려고 그림처럼 캡슐 패킷으로 변경하는데, 이 과정에서 새로운 IP 헤더가 추가된다. 그리고 추가된 헤더의 Destination Address 필드에는 COA(Care of Address)가 들어간다.

24.

네트워크 환경에서 임의의 호스트가 다른 호스트에 데이터를 전송하려면 수신 호스트의 IP 주소뿐만 아니라, MAC 주소도 알아야 한다. 수신 호스트의 IP 주소는 보통 응용 프로그램 사용자가 프로그램을 실행하는 과정에서 직접 입력하므로, IP 주소로부터 수신 호스트 MAC 주소를 얻는 작업이 추가로 필요하다. 이렇게 IP 주소로부터 MAC 주소를 얻는 기능은 ARP(Address Resolution Protocol)를 통해 이루어진다.

예를 들어, 호스트 A가 호스트 B의 MAC 주소를 얻으려면 ARP request라는 특수 패킷을 브로드캐스팅해야 한다. ARP request 패킷을 네트워크의 모든 호스트가 수신하지만, 관계없는 호스트는 패킷을 무시하고 호스트 B만 IP 주소가 자신의 IP 주소와 동일함을 인지한다. 따라서 호스트B는 ARP reply 패킷을 사용해 자신의 MAC 주소를 호스트 A에 회신한다.

25.

하드 디스크가 없는 호스트에서는 송신 호스트 IP의 주소를 보관할 방법이 없다. 따라서 LAN 카드에 내장된 MAC 주소를 매개변수로 하여 RARP 기능을 수행함으로써 자신의 IP 주소를 얻어야 한다. MAC 주소를 이용해 IP 주소를 제공하는 기능은 RARP(Reverse Address Resolution Protocol)이 제공한다.

IP 주소를 얻고자 하는 호스트에서는 MAC 주소를 매개변수로 하여 패킷을 브로드캐스팅한다. 보통 네트워크에는 RARP의 기능을 전담으로 수행하는 서버가 하나 이상 존재한다. 따라서 모든 호스트가 RARP 변환 요청을 받아도 해당 정보를 보관하고 있는 RARP 서버만 응답을 수행할 수 있다.

26.

ICMP에 의해 발생되는 주요 메시지는 프로토콜 헤더의 Type 필드 값에 따라 다음과 같이 구분된다. 대부분 오류에 관한 것이다.

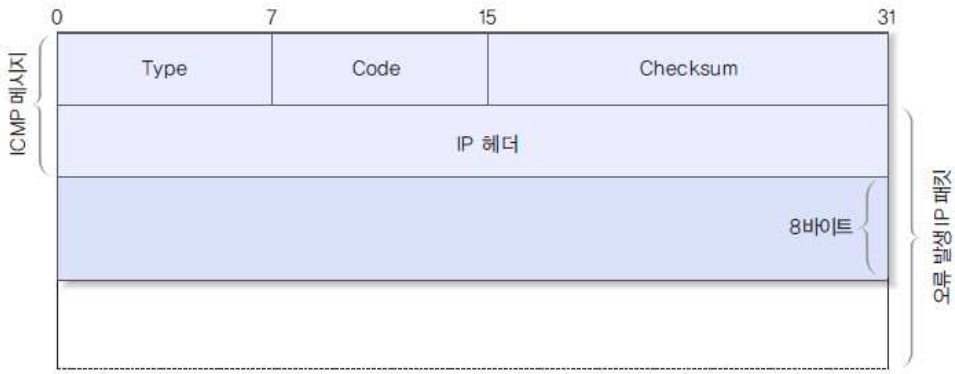
- ECHO REQUEST, ECHO REPLY : 유닉스(Unix)의 ping 프로그램에서 네트워크의 신뢰성을 검증하기 위하여 ECHO REQUEST 메시지를 전송하고, 이를 수신한 호스트에서 는 ECHO REPLY를 전송해 응답한다.
- DESTINATION UNREACHABLE : 수신 호스트가 존재하지 않거나, 존재해도 필요한 프로토콜이나 포트 번호 등이 없어 수신 호스트에 접근이 불가능한 경우에 발생한다.
- SOURCE QUENCH : 네트워크에 필요한 자원이 부족하여 패킷이 버려지는 경우에 발생한다. 예를 들면, 전송 경로에 있는 라우터에 부하가 많이 걸려 패킷이 버려지는 경우다. 이 메시지를 이용해 송신 호스트에게 혼잡 가능성을 경고함으로써, 패킷 송신 호스트가 데이터를 천천히 전송하도록 알릴 수 있다.
- TIME EXCEEDED : 패킷의 TTL(Time To Live) 필드 값이 0이 되어 패킷이 버려진 경우에

주로 발생한다. 기타의 시간 초과 현상에 의해 패킷이 버려진 경우도 이에 해당한다.

- TIMESTAMP REQUEST, TIMESTAMP REPLY : 두 호스트 간의 네트워크 지연을 계산하는 용도로 사용한다.

27.

ICMP 메시지(ICMP Message) 첫 줄의 4바이트는 모든 메시지에서 동일 한 구조를 보이지만, 이어지는 메시지 내용은 가변적이다. 단, 4바이트의 ICMP 메시지 내용 1을 포함해 총 8 바이트의 ICMP 정보는 반드시 포함한다.



ICMP 메시지 내용 1은 DESTINATION UNREACHABLE, SOURCE QUENCH, TIME EXCEEDED 등에서는 사용되지 않으므로 0이 채워지지만, ECHO REQUEST, ECHO REPLY, TIME STAMP REQUEST, TIME STAMP REPLY 같은 메시지에서는 특정 값이 주어진다.
1바이트의Type 필드는 메시지를 구별하며, Code 필드는 메시지 내용에 대한 더 자세한 정보를 알려준다. Checksum 필드는 ICMP 전체 메시지에 대한 체크섬 기능을 지원한다. ICMP의 주요 임무는 전송 오류 보고다. 오류 메시지를 전송할 경우에 ICMP 메시지 내용 2에는 오류 원인을 제공한 IP 패킷의 헤더와 이어지는 8바이트의 정보가 포함된다. 이 정보를 기초로 하여 IP 패킷 송신 호스트는 ICMP 오류를 파악하고 정정한다.

28.

호스트가 IGMP 메시지에 표시된 멀티캐스트 주소의 멤버임을 다른 호스트와 라우터에 알리기 위한 용도로 IGMP를 사용한다. 즉, 그룹에 가입하려면 해당 멀티캐스트 주소를 표기한 IGMP 보고 메시지를 전송해야 하는데, IGMP 헤더의 Group Address 필드에 가입을 원하는 멀티캐스트 주소를 기록한다. 멀티캐스트 라우터가 그룹에 속한 멤버 목록을 유효하게 관리하려면 IGMP 질의 메시지를 사용해 주기적으로 확인하는 과정이 필요하다. 개별 호스트가 자신의 그룹 멤버를 유지하려면 IGMP 보고 메시지를 사용해 IGMP 질의에

응답해야 한다. 라우터의 질의 메시지에 대해 호스트의 보고 메시지 응답이 이루어지지 않으면 그룹에서 탈퇴한 것으로 간주된다.

29.

IGMP 헤더 구조와 역할은 다음과 같다.

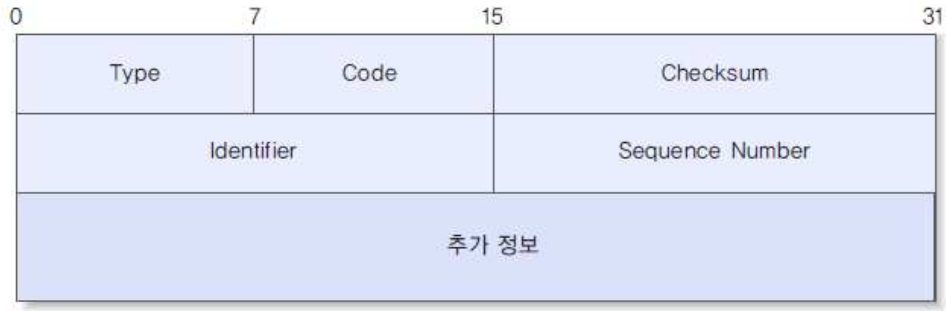


그림 8-10 ICMP 메시지 : 질의 메시지

- Version Number(버전 번호) : IGMP의 버전 번호로, 현재 버전은 1이다.
- Type(유형) : 크게 세 가지 값을 가질 수 있다. 0x11은 멀티캐스트 라우터가 전송한 질의 메시지며, 0x16은 호스트가 전송하는 보고 메시지다. 0x17은 그룹 탈퇴에 관한 메시지로, 특정 그룹에 소속된 마지막 멤버의 탈퇴와 관련된다. 이전 버전과의 호환성을 위해 0x12가 보고 메시지로 사용될 수 있다.
- Max Response Time(최대 응답 시간) : 질의 메시지에서만 사용된다. 질의에 대한 보고 메시지가 전송되어야 하는 최대 응답 시간을 나타낸다. 라우터는 이 값을 변화시킴으로써 탈퇴 지연(Leave Latency) 시간을 조율(Tune)할 수 있다. 탈퇴 지연 시간은 특정 그룹에서 마지막 호스트가 탈퇴한 시간과 라우팅 프로토콜이 이 사실을 인지한 시간의 차이다.
- Checksum(체크섬) : IP 프로토콜에서 사용하는 알고리즘과 동일한 방식을 사용하며, 오류 검출용으로 이용된다.
- Group Address(그룹 주소) : 질의 메시지는 0으로 채우고, 보고 메시지는 호스트가 가입을 원하는 그룹 주소를 표기한다. 특정 그룹에 관련된 질의 메시지에서는 해당 그룹의 주소를 표기해야 한다.