네트워크 프로토콜 분석

1. IP (Internet Protocol)

1.1 역할 및 특징

IP는 N/W 계층에서 TCP/IP 프로토콜이 사용하는 전송 메커니즘.

IP는 신뢰성X, 비연결형 프로토콜→ 최선의 노력(Best effort)을 하는 전달 서비스

Ef) 최선의 노력 → IP 패킷 오류 발생, 분실, 지연, N/W 내에서 혼잡 문제 발생할 수 있다는 의미

따라서 IP는 비연결형 프로토콜 → 패킷 독립적 처리, 목적지까지 다른 경로로 전달, 패킷 순서 뒤바뀜, 분실, 훼손

IP는 이 모든 문제 해결 위해 상위 계층 프로토콜에 의존함.

1.2 IP패킷 header format → 고정 부분(20bytes)과 가변 부분(Option)으로 구성

헤더가 중요한 이유 → 패킷의 기본 정보를 다루고 있음.

1. Version (4bits)	2. Header Length (4bits)	3. Type of Service	4. Total Length (16bits)	5. Identification (16bits)	6. Flags (3bits)	7. Fragment Offset (13bits)
8. TTL (8bits)	9. Protocol (8bits)	10. Checksum (16bit)	11. Source Address (32bits)	12. Destination Address (32bits)	13. Option	-

1) 버전: IP 버전을 나타냄 → 시스템 내 수행되고 있는 IP S/W에게 패킷이 버전 4 형식으로 되어있다는 것을 알려줌.

2) 헤더 길이: 헤더의 전체 길이를 4바이트 단위로 나타냄. 헤더의 길이는 최소 20bytes ~ 최대 60bytes

3) 서비스 유형: 현재 사용하지 않는 필드 → 중간 라우터에서 이 필드는 고려하지 않음, Service type 오류나도 상관 없다.

4) 전체길이: 패킷의 전체 길이(Header+Paylaod)를 바이트 단위로 나타냄. 필드의 길이가 16비트이므로 패킷 전체 길이는 2^16-1로 제한됨. 데이터링크 계층의 이더넷의 MTU는 1500Bytes이므로 제한이 있다. 따라서 20바이터에서 60바이트는 헤더이며, 나머지는 상위 계층으로부터 받은 데이터이다.

5) 식별: 이더넷의 MTU에 의해 쪼개진 단편화된 패킷이 재조립될 때 동일한 패킷이였다는 식별하기 위한 정보이다.

6) 플래그: 이 필드는 단편화를 위하여 사용됨. 편화 시 사용하는 필드이며, 패킷의 분할을 제어한다.

XXX	Don't Fragment (DF)	More Fragment (MF)

처음 비트 사용되지 XXX, 두번째 비트 플래그 값이 1이면 패킷을 단편화해서는 안 된다. 단편화 수행해야 하는데 이 비트가 설정되어 있다면 패킷을 폐기하고 ICMP 오류 메시지를 발신지 호스트에 보냄.

세번째 비트 플래그 값이 1이면 패킷의 마지막 단편이 아니라는 것을 알려준다. 0이면 마지막 단편 or 유일한 단편이다.

7) 단편화 옵셋: 전체 패킷 내에서 단편의 상대적 위치를 나타냄. 데이터 옵셋을 8바이트 단위로 나타냄.

8) TTL: 패킷에 의해 방문되는 최대 홉(라우터) 수를 제어하기 위해 사용되며, 패킷의 수명을 제한

발신지 호스트가 패킷을 보낼 때 이 필드에 숫자를 저장한다. 이 값은 대략 두 호스트 사이에 위치한 라우터 수의 2배이다. 패킷을 처리하는 각 라우터는 이 필드 값을 1씩 감소시키며, 만약 감소 후 값이 0이 되면 라우터는 패킷을 폐기함.

인터넷 내의 라우팅 테이블이 오염될 수 있으므로 이 필드가 필요함.

발신자가 패킷이 같은 네트워크 내에서만 전달되길 원한다면 이 필드 값을 1로 설정하면 된다.

9) 프로토콜: 상위 계층 프로토콜을 정의. IP패킷은 TCP, UDP, ICMP, IGMP와 같은 여러 종류의 상위 계층 프로토콜을 캡슐화.

10) 검사합: 패킷 전달 중에 발생할 수 있는 오류로부터 패킷을 보호.

검사합은 송신자에 의해 계산되고 패킷과 함께 전송 → 수신자는 검사합을 포함하고 있는 전체 패킷에 대해 같은 계산을 반복. → 결과가 만족되면 패킷은 받아들여지고 그렇지 않으면 폐기

11) 발신지 주소: 발신지 P 주소를 정의 → IP 패킷이 발신지에서 목적지까지 전달되는 동안 이 값 변해서는 안 된다.

12) 목적지 주소: 목적지 IP 주소를 정의 → IP 패킷이 발신지에서 목적지까지 전달되는 동안 이 값 변해서는 안 된다.

13) 옵션: 가변 부분에 해당되며 최대 길이는 40bytes. 패킷의 추가 정보를 포함하는 필드.

1.3 보안

IP Security 프로토콜을 사용해 IP 패킷을 보호할 수 있다. IP Security는 다음 2가지의 서비스를 제공한다.

1) 패킷 encryption: 기밀성 유지하기 위해 암호화 알고리즘과 키를 사용해 암호화

2) 데이터 무결성: 패킷 전달 과정에서 데이터가 변경되지 않았음을 확인. 만약 수신된 패킷이 데이터 무결성 검사를 통과하지 못하면 패킷은 폐기됨.

2. TCP (Transmission Control Protocol)

2.1 역할 및 특징

전송 계층에서 사용되는 프로토콜이며, 데이터 단위는 세그먼트이다. 신뢰성, 순서 보장, 하며 세그먼트 분실(손실) 시 재전송한다. TCP에서 핵심 → Seq. number와 Ack. Number를 사용한 Numbering system, 오류 제어, 혼잡 제어, 흐름 제어를 수행. 따라서 TCP는 Connection oriented and reliable protocol이다.

연결을 설정할 때는 3-way-handshake방식을, 해제할 때는 4-way-handshake방식을 사용한다.

2.2 번호화 시스템

1) 바이트 번호: TCP는 한 연결에서 전송되는 모든 데이터 바이트에 번호를 매긴다.

2) 순서 번호(Seq number): TCP는 전송하고자 하는 세그먼트에 하나의 순서 번호를 할당한다.

3) 확인응답 번호(ACK number): TCP는 자신이 바이트를 수신하였다는 것을 확인하여 주기 위해 확인응답 번호 이용한다.

2.3 세그먼트 헤더 형식

세그먼트 → 20~60Bytes 길이의 헤더와 위 계층으로부터 내려온 데이터로 구성됨.

1. Source Port (16bits)	2. Dest. Port (16bits)	3. Seq. Number (32bits)	4. Ack. Number (32bits)	5.Header length (4bits)	6. Reserved (6bits)
7. Control Flags (6bits)	8. Window Size (16bits)	9.Checksum (16bits)	10. Urgent Pointer (16bits)	11. Options	XXX

1) 발신지 포트 주소: 세그먼트를 전송하는 호스트에 있는 응용 프로그램의 포트 번호를 정의.

2) 목적지 포트 주소: 세그먼트를 수신하는 호스트에 있는 응용 프로그램의 포트 번호 지정.

3) Sequence Number: 세그먼트에 포함된 데이터의 첫 번째 바이트에 부여된 번호를 나타냄. → 목적지 TCP에게 세그먼트의 첫번째 바이트가 이 번호에 해당하는 바이트라고 알려준다.

TCP는 신뢰성 있는 연결 보장 위해 전달되는 각 바이트마다 번호를 부여.

각 TCP에서는 난수 발생기 이용해 초기 순서 번호를 만들며, 초기 순서 번호는 TCP마다 다르다.

4) Acknowledgment Number: 세그먼트를 수신한 수신 노드가 이어서 수신하고자 하는 바이트 번호 정의.

→ 수신 측에서 송신 측에게 Ack신호를 보냄으로 세그먼트를 잘 받았다는 의미로 Seq number+1 값으로 보낸다.

5) 헤더 길이: TCP 헤더 길이를 4바이트로 나타냄

6) 예약: 이 필드는 차후 사용 위해 예약된 6비트 필드.

7) 제어

URG	ACK	PSH	RESET	SYN	FIN

7.1) URG(Urgent): 2진수 1로 설정된 경우 순서에 상관없이 먼저 처리하라는 의미.

7.2) ACK(Acknowledgement): ACK세그먼트에서 사용되는 필드

7.3) PSH(Push): 비트가 1일 경우, 즉시 즉시 응용 프로그램으로 올리라는 의미.

7.4) RST(Reset): 수신 측에서 송신 측과 연결 해제 위한 control 비트

7.5) SYN(Synchronize): 최초 연결 설정 시(Syn. 세그먼트 and Syn. + Ack. 세그먼트) 사용되어진다.

7.6) FIN(Finish): 연결 종료 위한 control 비트

8) 윈도우 크기: 바이트 형태의 송신 TCP 윈도우를 나타냄. 이 필드의 길이는 16비트이므로 윈도우 최대 크기는 2^16-1.

윈도우 크기는 수신윈도우(rwnd: receiving window)라고 하며 수신 측에 의해 결정된다.

TCP는 흐름 제어 위해 자신의 버퍼 여유 크기를 지속적으로 통보한다.

9) 검사합: 헤더를 포함한 전체 세그먼트에 대한 오류를 검사하기 위한 필드. TCP에서 검사합은 필수 사항이다.

10) 긴급 포인터: 긴급 플래그가 1로 설정된 경우에만 유효한 이 필드는 세그먼트가 긴급 데이터를 포함할 때 사용된다.

이 필드에 있는 값과 Sequence number 더하며 세그먼트 Payload부분에 있는 마지막 긴급 바이트 번호를 알 수 있다.

11) 옵션: TCP는 최대 40Bytes까지 옵션 정보가 있다

.

2.4 TCP 연결

TCP는 연결 지향 프로토콜이다. 연결은 의무적이며, 가상 커넥션이다.

TCP의 연결 지향 전송은 연결 설정-데이터 전송-연결 종료 3가지 단계를 통해 이뤄진다.

2.4.1연결 설정

TCP에서 연결 설정은 3-way handshaking이라 한다.

1) 클라이언트는 SYN 플래그 비트가 1로 설정된 SYN 세그먼트를 전송. 이 세그먼트는 순서 번호 동기화가 목적이다.

→ SYN 세그먼트는 데이터를 전달하지 않지만 하나의 순서 번호를 소비한다.

2) 서버는 SYN과 ACK 플래그 비트가 1로 설정된 SYN+ACK 세그먼트를 전송한다.

이 세그먼트는 ACK Number를 포함하고 있으며, 수신 윈도우 크기인 rwnd를 포함한다.

→SYN+ACK 세그먼트는 데이터를 전달하지 않지만 하나의 순서 번호를 소비한다.

3) 클라이언트는 ACK 세그먼트를 전송한다.

2.4.2 데이터 전송

연결 설정 후 클라이언트와 서버는 양 방향으로 데이터와 확인응답을 전송할 수 있다.

데이터와 확인응답은 하나의 세그먼트로 전달될 수 있다 → 피기백

2.4.3 연결 종료

어느 쪽도 연결을 종료할 수 있지만, 일반적으로 클라이언트에서 종료를 시작함.

연결 종료를 위해 3단계 핸드셰이크 or 반닫기 옵션을 가진 4단계 핸드셰이크가 있다.

2.5 오류 제어

신뢰성 있는 서비스 제공 위해 TCP는 오류 제어 메커니즘 구현 → 오류 제어 메커니즘의 핵심은 세그먼트 재전송이다.

2.5.1 Stop and Wait ARQ

송신 측에서 1개 세그먼트 전송 후, 수신 측에서 Ack 받을 때까지 대기

2.5.2 Go Back N ARQ

손실된 세그먼트 이후 모든 세그먼트 재전송

2.5.3 Selective Repeat ARQ

손실된 세그먼트만 재전송

2.6 흐름 제어

송수신 측 사이 데이터 처리 속도 차이를 제어하기 위한 기법 → 데이터 처리 속도 조절해 수신 측 오버플로우 방지

2.6.1 Stop and Wait

전송한 세그먼트에 대한 Ack 받아야 그 다음 세그먼트 전송 가능

2.6.2 Sliding Window(슬라이딩 윈도우)

수신 측에서 설정한 윈도우 크기만큼 송신 측에서 Ack 없이 세그먼트를 전송하여 흐름을 동적으로 조절.

2.7 혼잡 제어

N/W 혼잡 피하기 위해 송신 측에서 보내는 데이터 전송 속도를 제어하기 위한 기법

2.7.1 AIMD(Additive Increase Multicative Decrease)

세그먼트 문제없이 도착한다면 윈도우 크기 1씩 증가. 만약, 전송을 실패 or 타임아웃 발생 시 윈도우 크기 절반 감소

2.7.2 Slow Start

세그먼트 문제없이 도착한다면 윈도우 크기 1씩 증가. 한 주기 지나면 미리 정해진 임계값 도달 시까지 윈도우 크기 2배씩 증가 → 지수함수 형태

임계값 도달 시 혼잡 회피 단계로 진입(1씩 증가)

2.8 재전송

2.8.1 3 duplicate ack

2.8.2 Time-out 발생

3. UDP (User Datagram Protocol)

3.1 특징

전송계층에 속하는 프로토콜. 데이터 단위는 데이터그램. TCP와 상반된 프로토콜이며 연결 설정 및 어떠한 Control 기능도 없다. 비신뢰성 및 비연결성으로 인한 비순서, 손실, 훼손 야기할 수 있다. → 재전송 메커니즘 존재 X

Best effort → 네트워크 상황에 따라 최선의 노력만 함.

UDP프로토콜은 TCP에 비해 헤더가 단순하며 상대방의 수신 응답 여부와 상관없이 전송하므로 전송 속도가 빠르다.

3.2 UDP 헤더 구조

Source Port (16bits)	Destination Port (16bits)	Total Length (16bits)	Checksum (16bits)

3.3 UDP Checksum

N/W를 통해 전송된 데이터가 변경되었는지 무결성을 검사하는 값.

IP의 Checksum 계산 방법과 동일 → 16bit 워드 단위로 모두 더함 → 1의 보수 취한 값이 Checksum

체크섬 UDP 헤더에 포함 → 수신 측에서 체크섬 계산 후 비교해 값 다르면 패킷 버림 → 재전송 XXX

IpV4에서 Checksum은 옵션 → IPv4에서는 필수.

3.4 UDP Flooding Attack

공격자는 UDP 특징을 악용하여 상대 시스템에 대량의 UDP 패킷을 보내 시스템 리소스 고갈시키는 공격 → N/W 성능 저하, 서비스 거부 상태 야기

4. ARP (Address Resolution Protocol)

4.1 역할 및 특징

ARP는 논리적 주소 및 식별자 역할을 하는 IP 주소를 통해 물리적 주소인 MAC 주소를 알 수 있는 방법이다.

ARP에서 핵심은 MAC 주소는 중복되지 않는 unique한 값이다. 또한 MAC Address는 NIC에 사용되는데 1개의 PC에는 복수의 NIC를 부착할 수 있다.

MAC Address는 총 48비트이며, 왼쪽부터 24비트는 제조사 식별정보이며 나머지 24비트는 unique한 고유번호이다.

4.2 ARP 헤더 구조 분석

1. H/W Type (16bits)	2. Protocol Type (16bits)	3. H/W Add. Length	4. Protocol Legnth	5. Operation
6. Sender H/W Add. (48bits for Ethenet)	7. Sender IP Add. (32bits for IPv4)	8. Target H/W Add. (48bits for Ethenet)	9. Target IP Add. (32bits for IPv4)	XXX

1) H/W Type: 이더넷 통신 시 값은 항상 1로 설정된다.

2) Protocol Type: 프로토콜 주소의 유형을 나타내며 IPv4는 0x0800으로 설정

3) H/W Length: MAC Address 길이를 bytes로 나타낸다. MAC Address주소는 48비트이므로 6임을 알 수 있다.

4) Protocol Legnth: 프로토콜 주소의 길이를 bytes로 나타냄

5) Operation: ARP의 구체적 동작을 나타냄

값	동작
1	ARP Request
2	ARP Reply
3	RARP Request
4	RARP Reply

6) 송신자 물리적 주소

7) 송신자 IP주소

8) 목적지 물리적 주소

9) 목적지 IP주소

4.3 ARP 동작과정

4.3.1 두 단말이 서로 같은 N/W에 있을 때

1) 호스트A는 호스트 B의 IP Address를 포함하는 ARP Request 패킷을 Broadcast

2) 호스트B는 자신의 IP Address를 인지하고, 자신의 H/W주소를 포함한 ARP Reply 패킷으로 응답.

3) 호스트A는 ARP캐쉬 내에 호스트B의 IP Address에 대응한 H/W Address를 매핑한다

4) 송신자는 수신자의 MAC주소를 확인하고 B와 통신을 시작한다

4.3.2 두 단말이 서로 다른 N/W에 있을 때

1. 호스트A는 호스트 B의 IP Address를 포함하는 ARP Request 패킷을 Broadcast

2. ARP패킷은 호스트A의 Gateway(라우터)로 전달된다.

3. 호스트A의 N/W의 라우터는 자신의 MAC주소로 수신자 라우터 MAC 주소로 라우팅

4. 호스트 B의 N/W의 라우터는 자신의 MAC 주소 확인하고 ARP Request를 Broadcasting

5. 호스트B는 자신의 IP임을 확인 후 다시 자신의 네트워크 라우터로 ARP Reply

6. 호스트B의 네트워크 라우터는 ARP테이블에 호스트B의 MAC 주소를 기록하고, 호스트A의 네트워크로 전송

5. ICMP (Internet Control Message Protocol)

N/W 계층의 프로토콜로서, IP와 함께 사용된다. IP 패킷 헤더 뒤에 포함되어 전달되는데 Ip패킷 헤더의 Protocol필드 값이 1이면 ICMP메시지를 전달한다.

호스트 or 라우터가 오류에 대해 보고를 하거나 비정상적인 경우에 관한 정보를 제공한다. Ex)라우터가 패킷을 폐기한 경우, 어떤 이유에서 해당 패킷이 목적지까지 도달하지 않은 경우

5.1 ICMP 주요 기능

1) 에코(echo) 요구(Request)와 응답(Reply): ping 명령어 → 패킷 송수신함으로써 N/W 상태 파악할 수 있다.

2) 목적지 도달 불가능: IP패킷이 수신자에게 전달될 수 없을 때

3) 시간 초과: IP 패킷의 TTL값이 만료되어 패킷이 버려지게 될 때

4) 리다이렉트: 후속의 IP패킷을 위해 더 좋은 경로를 알리기 위해 사용

5) 오류 및 상태 보고: 라우터, 네트워크 등의 상태 감지 및 보고

6) 네트워크 경로 추적: tracert명령어 → 목적지까지의 라우팅 경로 추적을 하기 위해 사용

5.2 ICMP 일반적 메시지 형식

ICMP 메시지는 IP패킷에 포함되어 있다.

IP header	ICMP header(32bits)	ICMP data (Optional)	IP Payload

Type	설명	ICMP Code	
0	Echo 응답(reply) 메시지	0	Echo reply
3	해당 목적지 도달할 수 없음 (Destination Unreachable)	0	Network 도달 불가
		1	호스트 도달 불가 – Host Unreachable (목적지 호스트 도달 불가)
		2	프로토콜 도달 불가 (목적지 호스트에서 특정 프로토콜을 사용할 수 없는 경우)
		3	포트 도달 불가 (traceroute등에서 목적지 호스트의 UDP 포트가 열려있지 않음)
		4	패킷 Fragment가 필요하나 DF 필드가 활성화되어 단편화를 못하는 경우, 라우터에 의해 반환
		5	Source 라우팅 불가
		6	목적지 Network 인식 불가
4	발신 제한	0	Source Quench (Source에게 패킷 전송 자제를 요구)
5	라우팅 변경(redirect) (라우터에게 현재 네트워크의 위치정보를 알림)	0	Redirect for Network
		1	Redirect for Host
		2	Redirect for ToS & Network
		3	Redirect for ToS & Network
8	Echo 요청(request) 메시지	0	Echo Request
9	자신이 라우터임을 알림	0	Router Advertisement
11	TTL이 "0"이 되었음을 송신 측에 알림. 시간 초과(Time exceeded)	0	패킷 전송중에 TTL 초과
		1	패킷 Reassemably중 TTL 초과

Type number가 3이므로 목적지를 찾을 수 없다는 걸 의미하는데 Code가 3이므로 해당 포트에 도달하지 못한 걸 알 수 있다.

6. DHCP (Dynamic Host Configuration Protocol)

6.1 개념 및 특징

DHCP 서버가 Host에게 IP 주소 DNS 주소, 서브넷 마스크, Gateway 주소를 일정 기간 동안 동적으로 할당하기 위한 프로토콜.

즉, TCP/IP에 접속하는 모든 컴퓨터는 자신의 IP주소, 라우터 IP주소, 네임 서버 IP주소, 서브넷 마스크를 알아야 한다.

따라서, DHCP는 처음 부팅되는 컴퓨터에게 네트워크 정보를 전달하기 위해 설계된 클라이언트/서버 프로토콜이다.

6.2 DHCP 패킷 형식

1. Operation code (8bits)	2. H/W Type (8bits)	3. H/W Length (8bits)	4. Hop count (8bits)
5. Transaction ID (32bits)	6 Number of seconds (16bits)	7. Flags (16bis)	8. Client IP Address (32bits)
9.상대방 IP Address (32bits)	10. Server IP Address (32bits)	11. Gateway IP Address (32bits)	12. Client H/W Address
13. Server name (64bytes)	14. boot filename (128bytes)	15. option (64bytes)	XXX

1. Operation code: Request 1, Reply 2

2. H/W Type: 물리 네트워크 유형을 나타냄. 이더넷의 경우 값 1

3. H/W Length: 물리 주소 길이를 바이트 단위로 나타냄. 이더넷의 경우 값 6

4. Hop count: 패킷이 전달될 수 있는 최대 홉 개수

5. Transaction ID: 클라이언트에 의해서 설정되며, 요청에 대한 응답을 확인하기 위하여 사용됨.

6. 초 단위 경과 시간: 클라이언트가 부팅된 후 경과된 시간을 초 단위로 나타냄.

7. 플래그: 16비트 중 맨 왼쪽 비트만 사용되고 나머지 비트들은 0으로 설정

8. 클라이언트 IP Address: 클라이언트 IP 주소를 나타낸다. 만약 클라이언트가 이 정보를 가지고 있지 않다면 모두 0으로 저장

9. 상대방 IP Address: 이 필드는 요청을 받은 서버에 의해서 Reply 메시지에 기록됨

10. 서버 IP Address: 서버는 이 값을 Reply 메시지에 기록

11. Gateway IP Address: 라우터 IP 주소이며, 서버에 의해 Reply 메시지에 기록됨.

12. Client H/W Address: 클라이언트의 물리 주소를 나타냄.

13. Server name: 서버가 Reply 메시지에 기록하는 선택 항목.

14. boot filename: 서버가 Reply 메시지에 기록하는 선택 항목.

15. option: 네트워크 마스크 or 기본 라우터 주소와 같은 추가적 정보를 전달할 때 사용. 이 필드는 Reply 메시지에서만 사용됨.

6.3 DHCP 동작 방식

6.3.1 DHCP Discover

클라이언트가 네트워크를 통해 통신하기 위해 네트워크 정보가 필요하므로 Broadcast로 DHCP Discover 메시지 보냄

6.3.2 DHCP Offer

서버는 할당되지 않고 남아있는 네트워크 정보를 클라이언트에게 보냄.

6.3.3 DHCP Request

클라이언트는 서버가 보내준 네트워크 정보를 사용하겠다는 메시지로 Request 보냄.

다른 DHCP서버로부터 Offer메시지를 수신했을 가능성 존재. 따라서 확인 절차로도 볼 수 있음.

6.3.4 DHCP Ack

DHCP서버는 클라이언트에게 Unicast로 메시지 보내며 마무리.

6.4 DHCP 동작 과정

DHCP 클라이언트와 서버는 동일 망에 있을 수도 있고 서로 다른 망에 있을 수도 있다.

6.4.1 동일 네트워크

일반적이지 않지만 관리자는 클라이언트와 서버는 동일한 망에 놓을 수 있다.

1) DHCP 서버는 UDP 포트 번호 67을 열고 클라이언트로부터의 요청을 기다린다.

2) 부팅된 클라이언트는 UDP 포트 번호 68 열기 명령을 수행.

발신지 포트 번호 68, 목적지 포트번호 67을 가지는 UDP 데이터그램으로 캡슐화. → 그 후 UDP 데이터그램은 IP패킷으로 캡슐화.

클라이언트는 현재 자신의 IP주소와 서버의 IP주소를 모르는 상태이기 때문에 Broadcast

3) DHCP 서버는 발신지 포트 번호 67, 목적지 포트 번호 68로 지정한 뒤 Unicast or Broadcast로 클라이언트에게 보냄.

DHCP 서버의 Reply가 Unicast 형태가 될 수 있는 것은 서버가 클라이언트의 IP주소를 알고 있기 때문이다.

6.4.2 서로 다른 네트워크

클라이언트와 서버는 서로 다른 네트워크에 있을 수 있다.

문제점 존재 → 클라이언트가 보내는 DHCP Request는 서버의 IP주소를 모르기 때문에 Broadcast.

클라이언트 네트워크 라우터는 Broadcast IP패킷을 다른 네트워크로 통과하지 못 하도록 폐기.

문제점 해결 → 라우터가 중계자로 사용될 수 있다. 이 경우 라우터를 **중계 에이전트**라고 부른다.

1) 클라이언트 측 중계 에이전트는 DHCP 서버 IP 주소를 알고 있으며, 포트 번호 67로 들어오는 Broadcast 메시지를 기다리고 있다.

2) 만약 중계 에이전트가 이러한 패킷을 수신하게 되면 이 패킷은 라우터들에 의해 라우팅 되며 DHCP 서버에 도착.

3) 클라이언트가 보낸 Request 메시지에 있는 필드 중 하나가 중계 에이전트의 IP 주소를 정의하고 있기 때문에 DHCP 서버는 중계 에이전트로부터 온 메시지임을 알게 된다.

4) 그 후 응답을 받은 중계 에이전트는 DHCP 클라이언트로 이를 전송한다.

6.5 DHCP의 UDP 포트

서버는 잘 알려진 포트67을 사용, 클라이언트도 잘 알려진 포트 68 사용

6.6 에러 제어 - Request or Reply 메시지가 분실 or 손상 경우?

DHCP는 UDP를 사용하는데 UDP는 에러 제어 기능을 제공하지 않음.

따라서 DHCP의 에러 제어는 다음 2가지 정책을 통해 이뤄진다.

→ DHCP는 UDP Checksum을 사용할 수 있다.

→ DHCP 클라이언트는 DHCP Reply를 수신하지 못하면 → 타이머 정책(랜덤 변수 사용)과 재전송 정책 사용

6.7 설정

DHCP는 정적 및 동적 주소 할당 모두 가능하다.

6.7.1 고정 주소 할당

DHCP 서버는 물리 주소와 IP주소를 매핑한 동적 DB 가짐.

6.7.2 동적 주소 할당

DHCP 서버는 이용 가능한 IP주소를 DB로 가지고 있음.

DHCP 클라이언트가 IP주소를 요청 → DHCP 서버는 DB에서 활용 가능하고 사용되지 않는 IP주소를 일정기간 동안 할당.

1) DHCP 클라이언트가 DHCP 서버에 요청을 전송하면, DHCP 서버는 먼저 정적 DB를 찾는다.

정적 DB에 과거 요청된 물리 주소가 있으면, 물리 주소에 매핑된 IP주소를 클라이언트에게 반환.

2) 정적 DB에 요청된 물리 주소가 없으면, 서버는 활용 가능한 IP 주소를 클라이언트에게 할당. → 동적 DB에 물리 주소를 추가.

3) 동적 측면에서 DHCP는 임시 IP주소를 제한된 시간 동안만 제공. → DHCP 서버는 이를 일정 기간 동안 임대(leasing)하는 것.

임대 기간이 끝나게 되면, 클라이언트는 이 주소 사용을 그만 두거나 or 임대를 새로 요청해야 함.

서버는 이런 새로운 요구에 동의 or 동의하지 않을 수 있다. 서버가 동의하지 않으면 클라이언트는 이 주소 사용을 그만둬야 한다.

7. SNMP (Simple Network Management Protocol)

7.1 개념 및 특징

네트워크에서 장치들을 관리하기 위한 기반 구조 프로토콜.

SNMP는 관리자와 에이전트 개념을 사용 → 관리자는 SNMP 클라이언트 프로그램을 수행하는 호스트이며, 에이전트는 SNMP 서버 프로그램을 수행하는 라우터이다. → 호스트인 관리자는 보통 라우터나 서버인 에이전트를 제어하고 감시.

1) 에어전트(라우터)는 DB에 성능 정보를 저장하고 관리자(호스트)는 이 DB에서 값을 읽는다. 예를 들어 라우터는 수신된 패킷 수와 전달된 패킷 수를 저장하는데, 관리자는 이 값을 얻어 비교하며 라우터가 폭주 상태인지 아닌지 알 수 있다.

2)관리자는 또한 라우터로 하여금 특정한 동작을 행하도록 할 수 있다. 라우터는 주기적으로 재시동 계수기 값을 검사하여, 스스로 재시동 해야 하는 때를 알 수 있다. 만약 계수기 값이 0이면 라우터는 스스로 재시동 해야 한다. 관리자는 이 기능을 사용하여 언제라도 라우터를 원격으로 재시동할 수 있다. 관리자는 단지 계수기 값을 0으로 만드는 패킷을 전송하면 된다.

3) 라우터는 네트워크 환경 관리 과정에도 기여할 수 있다. 라우터에서 수행되는 서버 프로그램은 네트워크 환경을 검사하며 관리자에게 경고 메시지(Trap)을 송신할 수 있다.

7.2 관리 구성요소

SNMP는 관리 작업을 수행하기 위해 아래의 2가지 프로토콜 MIB(Management information base)와 SMI(Structure of management information)를 사용한다.

1) SNMP→ 관리자와 에이전트 사이에 교환되는 패킷 형식을 정의하며 → 관리자와 에이전트 간 통신 프로토콜

2) MIB(Management information base) → 하나의 장비가 가지는 정보 객체의 집합

3) SMI(Structure of management information) → MIB 정보가 어떻게 정의되어야 하는지 기술한 정보

7.3 SNMP Message

7.3.1 SNMP Get

N/W 장비 기능 및 설정 정보 확인 위해 요청하는 메시지 타입

1) GetRequest: 한 개의 정보만 수집할 때 사용

2) GetResponse: Agent로 응답

3) GetNextRequest: 연속적으로 다수 존재하는 정보를 한 번에 가져오기 위한 메시지

4) GetBulkRequest: 정보를 묶음 통째로 가져오기 위한 메시지

7.3.2 SNMP Set

SNMP 이용해 N/W 장비의 설정을 변경하기 위한 메시지 타입

7.3.3. SNMP TRAP:

이상 상황 발생 시 N/W 장비가 관리 솔루션에게 즉시 전송하는 메시지를 보낼 때 사용하는 타입

Trap은 괸리자에게에게 메시지를 능동적으로 전송함으로써, 장비 모니터링에 있어서 놓칠 수 있는 부분을 보완하는 역할을 함.

7.4 UDP 포트

Get과 Set은 UDP 161번 포트를 사용. Trap은 UDP 162번 독자적 포트 번호를 사용하여 정보 교환 통로를 따로 두고 있음.

7.5 Object ID(OID)

N/W 장비의 기능 및 설정을 Object ID를 통해 구별함.

Ex) N/W장비의 CPU 사용률을 OID(1,3,2,6,4...)로 확인

→ 각각의 숫자에는 모두 의미가 있으며 숫자들을 쭉 따라 내려가면 N/W장비에 대한 정보가 나옴. 해당되는 OID로 SNMP GetRequest 하면 N/W 장비는 해당 OID에 적재된 현재 값을 Response 한다.

7.6 보안

SNMPv3는 이전 버전에 보안과 원격관리라는 2개 특성을 추가되었다.

N/W 장비 정보 값을 알기 위해 비밀번호를 입력해야 한다. 비밀번호는 인증과 암호로 구성.

인증 값은 해시 알고리즘 사용, 암호는 AES 알고리즘 사용

또한 관리자가 장치가 위치한 곳에 같이 있지 않아도 되는 보안 측면의 원격 설정 허용