

4/27 Routing Algorithm

IPv4 NAT

gateway router에서 패킷이 나갈 때 source ip를 gateway router ip로 변경하고, 응답이 왔을 때 변환 작업을 해서 다시 source ip에 돌려줘야 한다. 이때 port#n을 가지고 식별한다.

gateway router는 여러 개의 ip를 (인터페이스가 여러개일 수밖에 없음) 가질 것

레이어 디자인이 무너진다는 점. 이거는 원래 그 단 레이어 작업자가 그 위아래 레이어를 생각 안하게 만들려고 한건데.... 그걸 뜯어가지고 고쳐야 한다는게 이상한 것임...

70년대에 IPv4가 디자인되고 반 세기가 지나면서 설계상의 주소 공간 부족과 보안 문제가 불거졌다. 이 프로토콜을 버릴 수 없어서 모든 게 다 임시방편으로 덕지덕지 붙여서 해결하고 있는데, 결국 새로운 방식으로 갈아타야 하고 그 대안 중 하나가 IPv6.

IPv6이 1996년에 디자인되어 완성되었는데 이것도 거의 뭐 30년이 다 되어가는 상황이라, 어떤 요구사항이 있을지 알고 이걸 쓰냐(따라서 Future Internet Architecture로 가야한다)라는 입장이 있어 논란이 있음. 하나의 생태계이기 때문에 선불리 바꿀 수 없는 문제가 있어 유연하게 대응할 수 없는 문제가 있어서 아예 FIA로 가자는 것임.

- 모든 라우터를 동시에 바꿔야 함
- 기술적인 문제가 아니라 생태계적인 문제

DHCP

Dynamic Host Configuration Protocol

```
IP 192.168. 1. 47
MSK 255.255.255. 0
ROU 192.168. 1. 1 == Gateway
DNS 192.168. 1. 1
<- local name server의 ip주소: 예를 들어 지금 naver.com으로 가면
naver.com의 ip를 어딘가에 물어봐야 하는데 그걸 물어볼 서버가 dns.
그게 router 안에서 동작하고 있는 것
DHCP 또한 router에 있다.
```

어디서 접속하든 ip주소를 자동으로 할당하는 프로토콜

host는 어떻게 ip주소를 얻는가?

1. 수동 - Hard-Coded
2. DHCP - Plug-And-Play

- a. 목적: 어디서 접속하더라도 ip주소를 자동으로 할당해, 인터넷을 사용할 수 있게 함
- b. 고정IP정책을 사용했을 때에는 전체 사용자 수만큼 필요하지만, DHCP를 쓰면 동접자 수만큼 필요함
- c. 그러므로 IP주소를 회수할 수 있어야 함

동작과정

- 어떤 서브넷에 클라이언트가 들어오면, 클라이언트는 아무것도 알 수 없는 상황임 (누가 나한테 ip주소를 줄 수 있는지, ip주소는 무엇인지, ... 이걸 아무것도 모름)
- DHCP Discover MSG를 서버에 보낸다.
 - 이때 src는 0.0.0.0.68 (포트번호 고정)
 - 목적지는 255.255.255.255:67 (포트번호 고정, 모든 address bit가 1인 경우면 broadcast임), 이는 서브넷에 있는 모든 멤버가 이 메시지를 받으라는 의미
 - transaction number를 지정해서 보냄 (123이라고 가정)
- broadcast되면, 이 메시지를 DHCP 서버만 의미있게 받아들이고 나머지는 무시한다. (67번 포트를 연 곳은 DHCP 서버밖에 없으므로 나머지는 drop됨)
- DHCP서버는 DHCP offer를 보냄. src: **지정된 IP주소**, dest:255.255.255.255.68, 같은 원리로 클라이언트만 해당 offer를 의미있게 받아들임. lifetime이라는 회수를 위한 제한시간 존재. tx=123
- 클라이언트는 DHCP request를 보내서 이 offer를 수락함. src=0.0.0.0.68, dest=255.255.255.255.67.으로 보내되, 수락한다는 의미에서 tx#을 ++해서 보냄.
- DHCP 서버는 DHCP ACK를 보내고, offer에는 지정해준 ip, router, dns ip같은 것들이 다 들어있었기 때문에 DHCP offer를 그대로 사용하게 된다.

DHCP서버가 여러 개라면 offer가 여러 개 올 수 있고, 그 중에 하나를 고르게 된다. 때문에 DHCP ACK에 ip주소를 지정해서 보내주는게 아니라 offer에 있는 걸 사용해서, 나머지가 선택 안되었다는 사실을 간접적으로 알 수 있도록 한다.

실제로는 DHCP 프로세스를 gateway에서 작동시킨다.

```
client → dns query → dns → gateway router → ... → NAVER → ... → client
```

사실 router가 forwarding만 해야 하는데 검사검사 붙여놓은 것. 사실 DHCP, DNS, NAT, (firewall)는 router computer의 Application 단에서 돌고 있다.

인터넷 서비스 회사에서는 회선을 줄 때 ip주소 하나만을 주는데, 만약 집에서 공유기를 사용해서 여러 대의 디바이스에서 인터넷을 사용하는 상황이라면, 공유기에서 DHCP, DNS, NAT, firewall이 다 동작하고 있는 것임. 즉 공유기는 일종의 gateway router이다. 또, NAT는 중첩될 수 있다는 것을 유추할 수 있다.

IP fragmentation, and reassembly

maximum transfer size(MTU)는 각 link마다 다른데, 잘 가다가 특정 link의 MTU가 감소한다면? 쪼갬다. MTU 사이즈에 맞는 독립즈의 size frame으로 바뀌어서 보내는데, packet이 분리되어서 가고 link를 빠져나와 다시 assemble하여 전달된다.

만약 4000byte짜리 packet이 MTU가 1500인 link를 만났다면, 실제 data는 3980byte이다. 이 패킷은 3개로 쪼개야 하는데, 다음 세 개의 header를 이용해 다음과 같이 쪼갬다.

16bit identifier

frag.flag = 1, 0 (1==fragmented, 뒤에 frag 더 있음)

fragment offset = (첫 번째 시작 부분의 전체 패킷의 몇 번째인지)

```
length = 1500, ID = x, fragflag = 1, offset = 0    <- header + 1480 bytes of data field
length = 1500, ID = x, fragflag = 1, offset = 185  <- offset = 1480/8 (bit 3개 줄이려고)
length = 1040, ID = x, fragflag = 0, offset = 370  <- header + 1020 bytes of data field
                                                    AND offset = 2960/8
```

근데 이거 중간 거 날아가면 어떡함? network 단에서는 reassemble이 안 되고 날아감. 나중에 TCP가 해결하게 된다.

ICMP

IP packet의 header, data가 있는데 data는 거의 TCP segment(사용자가 보내고자 하는 message)일 것인데, ICMP는 이런 사용자 데이터 말고 네트워크에서 증상을 알기 위해 controll message를 보내는 프로토콜임. 예를 들어 포트번호가 맞지 않아 drop된 데이터나 TTL이 0이 되었을 때 src에게 report한다.

- network-layer “Above” ip라고 불리는데, ip Header이 아니라 datagram 안에서 운반되기 때문이다.
- Type, Code가 있는데, 예를 들어 11, 0이면 TTL Expired이다.

IPv6

결국 해결해야 하는 문제, 과도기 필요.

IPv4 datagram 안에 IPv6 datagram을 넣어서 보내주면서 과도기를 보내야함. 해석을 하냐 못하냐의 의미는 header format을 이해하냐 못하냐의 문제기 때문에... (tunneling)

Routing Alg.

routing은 결국 forwarding table lookup에 지나지 않는다(Longest Prefix Matching).
forwarding table을 그러면 어떻게 만듭?

- 모든 라우터에 대한 그래프가 존재한다고 생각하고 접근 - Link State
 - 모든 링크에 자기 link state를 broadcasting해서 연결 정보를 받아 그래프 생성
 - 다익스트라, 경로 복원
- 이웃에 대한 그래프만 존재한다고 생각하고 접근 - Distance Vector
 - 벨만 - 포드, DP