

Network Layer

Network layer : host to host, 한 호스트에서 다른 호스트로 라우팅(경로설정+경로결정), 망구조 네트워크에서 최적의 데이터 경로를 설정함 /IP

목적/기능

1. Packetizing : 모든 layer별로 기능하는 것. Segment를 받아서 Network layer에서 사용할수 있는 데이터 단위로 바꿔준다.(인캡슐레이션, 디캡슐레이션), 헤더를 붙여서 패킷으로 만든다.

원리 : 네트워크 레이어는 철저하게 payload를 까보지 않는다는 철칙이 있다.(편지를 보낼 때 우체국에서 까보지 않는다.) 그러나 까봐야 하는 경우도 있을수도?

2. Routing and forwarding : 서울->부산 두가지 일을 해야함

Routing (빠른길 찾기)

- 패킷이 전달 될 때에 라우터는 주변의 네트워크 상황을 고려하여 패킷을 전송한다. 가장 패킷의 이동이 원활한 네트워크로 패킷의 이동경로를 정한다.

Forwarding (보내기)

- 패킷이 라우터에 들어 오게 되면 적절한 출력 링크로 패킷을 전송하는 작업을 의미한다. 라우팅과정을 통해서 경로가 결정되면 포워딩하면서 적절히 전달하는 작업을 수행한다.

라우팅 알고리즘. 테이블 형태로 메모리에 저장해 둔다.

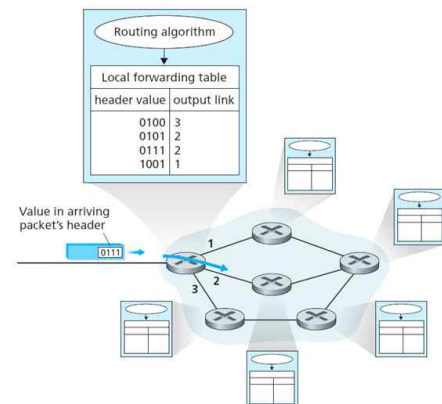


Figure 4.2 • Routing algorithms determine values in forwarding tables

IP Protocol

IP 프로토콜 : 인터넷에서 사용된다. 실제로 데이터를 실어 나르는 트럭이나 버스 역할.

Addressing convention 주소를 쓸 때의 약속.

Datagram format : 형식 규정

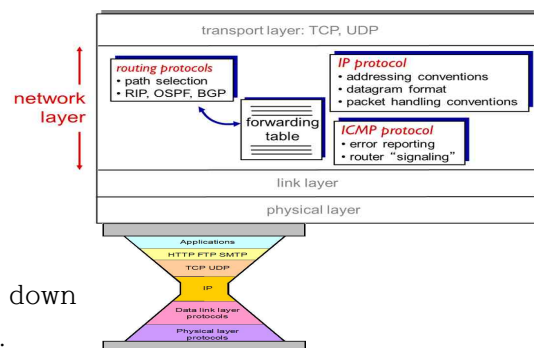
Routing protocol : 경로의 경우의 수를

알려주는 프로토콜(라우팅 관련 정보) -> 서로 정보를 교환해서 라우터가 경로를 결정

ICMP protocol : 라우터들을 감시하는 체계

인터넷이 성공한 이유 : ip로 통일되었다. 복잡도 down

Hourglass : 모래시계 형태. 인터넷 성공의 핵심.

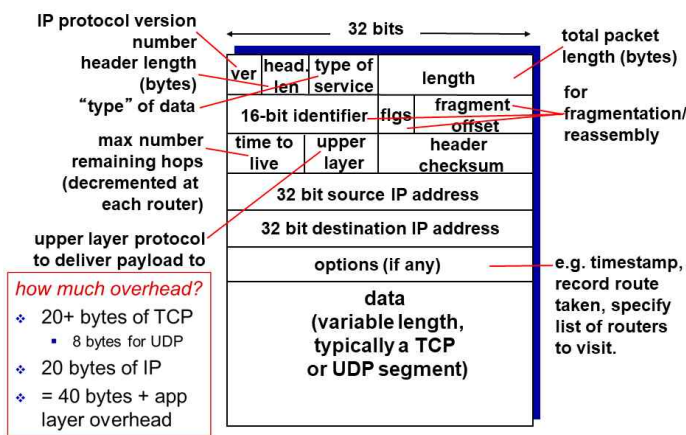


Packet switching : 버퍼 충분히 확보, congestion 발생 최대한 피해야 함

IP의 멋진 점

1. 하는 일이 없다. 최대 강점. 복잡하지 않고 simple하다.
2. Unreliable : 오류 투성이
3. Connection-less : Connection을 맺을 필요가 없다
4. Best-effort : 가능한 한 열심히 하겠습니다 충성
5. Out-of-sequence : TCP에서 out of order발생 이유 : 하위 layer인 IP에서 순서 뒤바뀌어서 보내서 그런거임

IP Field



1. Version : ip프로토콜의 버전 ipv4/6
2. Header length : 일반적으로 20byte. 5로 적혀 있을거임. 4(1byte=4bit)를 곱해라
3. Type of service : 옛날에 Type of service를 1byte로 사용함
-> DSF :
(-DSCP : 데이터의 가치 판단
-ECN : 네트워크 상태 (congestion) 명시적으로 알려주는 flag 실제로 사용 별로 안된다.)
4. Length : 패킷 전체의 길이

6. Time to live : 라우터 하나를 통과하는 것(한 홉을 지난다), 몇 홉을 지날 수 있는지에 대한 지표. 패킷은 일정 홉을 지나면 지워진다.
7. Upper layer : carry하고 있는 데이터의 유형(TCP/UDP)
8. Header checksum : 다음 라우터가 감지할 수 있도록 오류 방지 코드
9. Option : header length를 보고 option이 몇 개가 붙어있는지 알 수 있다.
(Timestamp : 특정 라우터를 지나온 시간, record route : 지나왔던 경로를 쓸 수 있다. Specify : 어디를 꼭 거쳐라 명시할 수 있다.)
Option을 제외하고 20byte이다.

Fragment(segment와 비슷. 그러나 세그먼트보다 좀 막짜르는 경향) : 하위에 physical layer가 실어 나를 수 있는 데이터 길이의 최대 길이에 맞춰 데이터를 자른 IP의 데이터 단위. 작은 단위(fragment)마다 헤더를 붙여준다.-> 전체 길이가 증가

10. 16-bit identifier : 라우터에서 그리고 최종 목적지에서 fragment들을 reassemble(재조합)한다. 이 때 fragment들이 하나였다는 것을 명시하기 위해 ID가 필요하다.
11. Fragment Offset : 첫 fragment에서 몇 바이트 뒤에서 시작되는 fragment인지 알려준다. 앞의 fragment의 길이를 명시해 두었다.
12. Flag : MF : fragment의 마지막 패킷이 아니다. 맨 마지막 fragment만 0
DF : 더 이상 잘라질 수 없을 때의 boolean

IP 주소체계

라우터 장비의 인터페이스에 할당된 32비트 주소. 같은 시간에 다른 인터페이스에 할당 불가능. 동시에 unique해야 한다. 만약 같은 주소가 동시에 할당되면 충돌 일어난다.

주소를 지리적 기반을 통하여 할당.

IP주소. 4byte, 네트워크 계층의 주소 체계

117.16.66.64 -> (network address)117.16.01000010.01000000(host address)

host address가 전부 11111111..이면 broadcast

서브넷 마스크

IP 주소와 1:1매칭하여 1인 부분은 network address, 0인 부분은 host address

255.255.254.0 2진수화 ->11111111.11111111.11111110.00000000

게이트웨이 : 맨 처음 만나는 라우터의 주소. 117.16.43.254처럼 보통 254로 끝난다.

Dotted-decimal-notation : 실제로 decimal이지만 binary수를 십진화해서 얻음 그리고 점을 찍음. 점 십진화

ex) 5bit 주소체계로 네트워크당 8개 host 배정할 때

앞의 2bit network address, 뒤의 3bit host address

서브넷 마스크 : 11000

address 01100 : 01(네트워크) 100(호스트)

네트워크 이름 : 01000, 네트워크 이름이므로 사용 불가능

01111은 broadcast 주소(local broadcast) (host address가 모두 1)

ex) 내 IP 주소 : 117.16.69.24/21(21 : 네트워크 주소의 비트 수)

서브넷 마스크 : 11111111.11111111.11110000.00000000

1.Network의 첫 번째 주소

2.Network의 마지막 주소

3.Network의 총 주소 수

IP Address의 할당 방법

1. Manual 할당(ex PC)

임의로 정해서 주소 할당시킨다.

2. Dynamic 할당(ex 노트북)

DHCP 프로토콜이 주소를 자동으로 할당 시켜준다.

필요할 때마다 IP주소를 할당받는다.

Manual IP configuration을 피할 수 있다. 모든 cpu를 일일이 다 하기에 너무 힘들다.

IP주소는 location과 연관이 있어서 장소에 따라 바뀌는 것이 유용하다.

왜 우리학교 pc는 고정식 IP주소를 사용할까? -> 문제 발생했을 때 추적 용이.