Computer Communication Networks

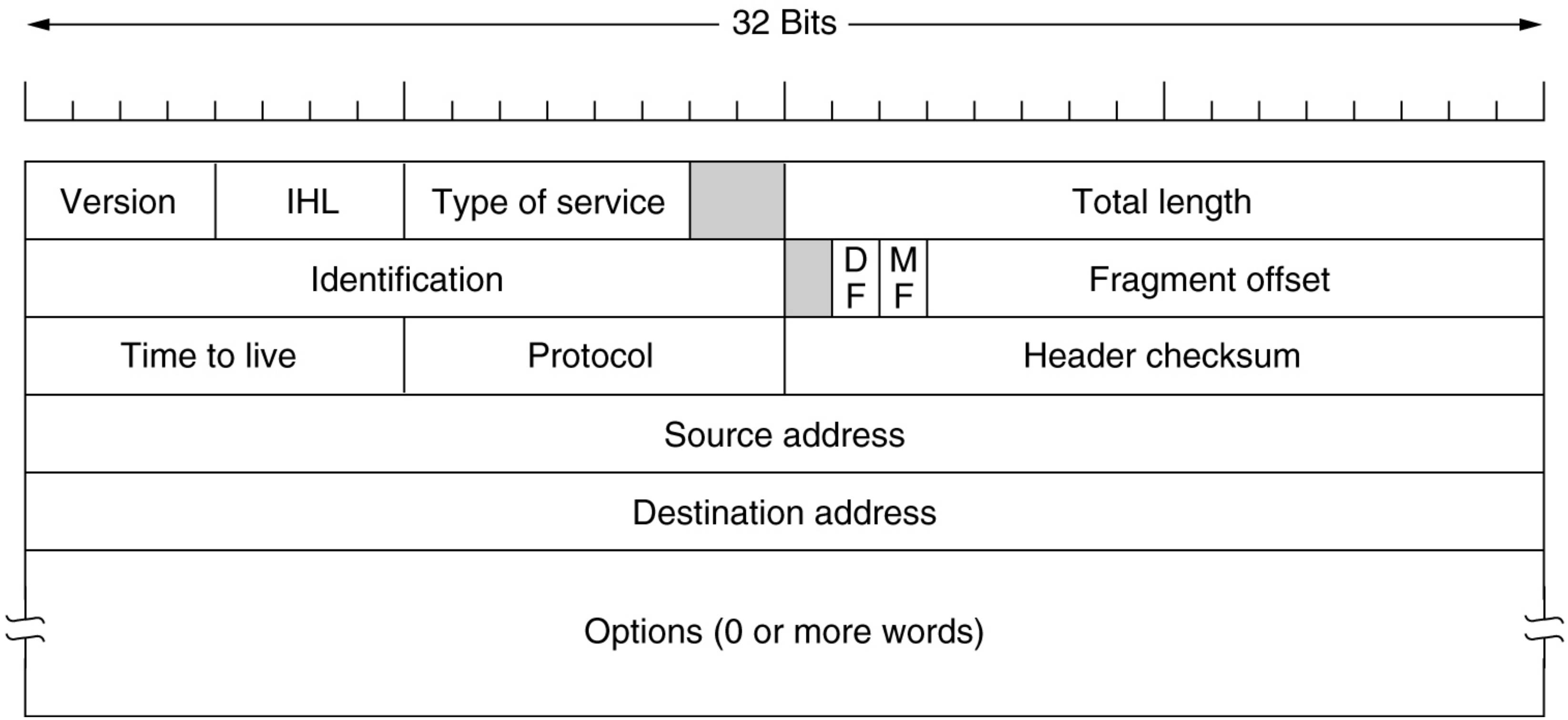
Internet Protocol

The Internet Protocol

- Service provided by link layer
  - frame delivery
    - point-to-point link
    - shared medium
- Service provided to transport layer
  - packet delivery
    - addressing and routing
  - best effort
    - lost, duplicated, reordered, corrupted

IP header

- IPv4



32 Bits

Version
IHL
Type of service
Total length

Identification
D F
M F
Fragment offset

Time to live
Protocol
Header checksum

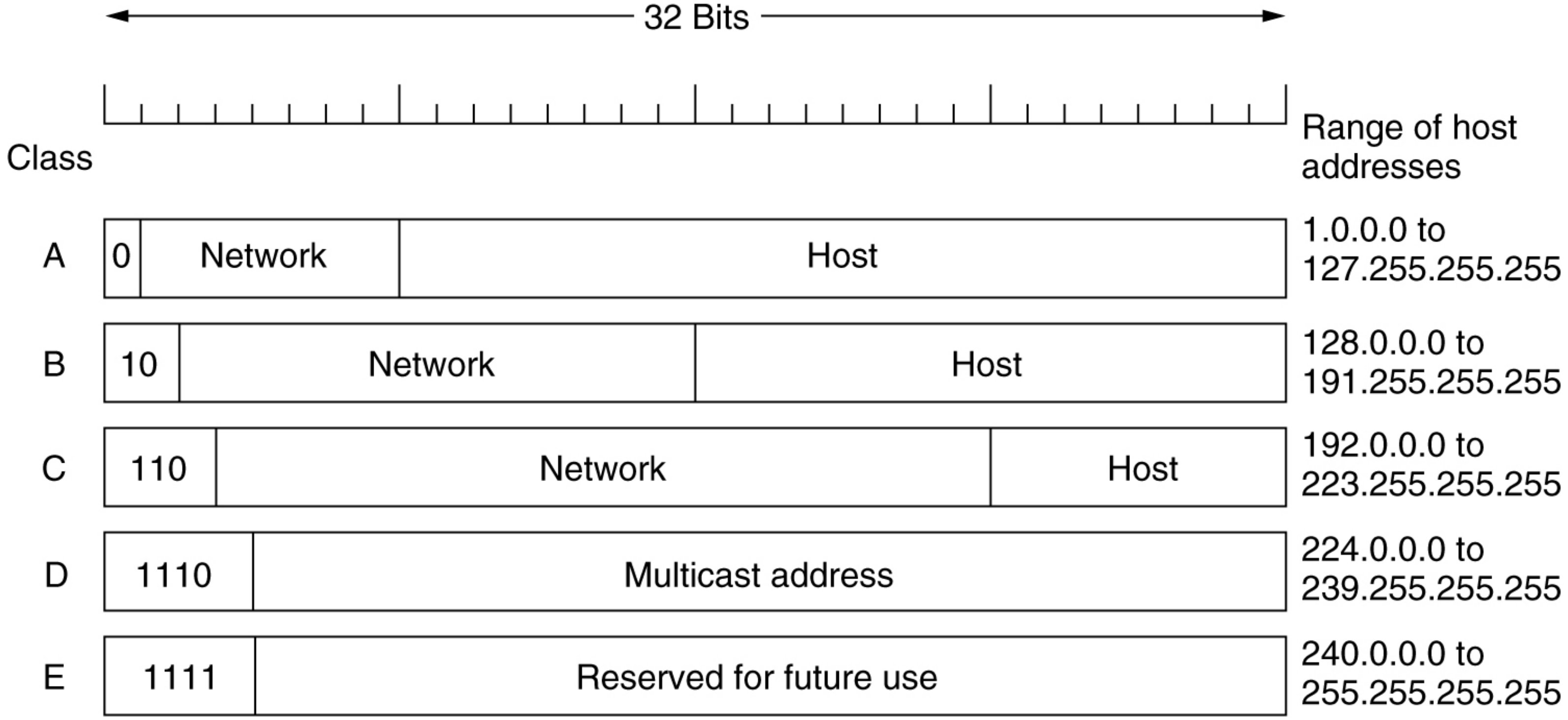
Source address

Destination address

Options (0 or more words)

IP address

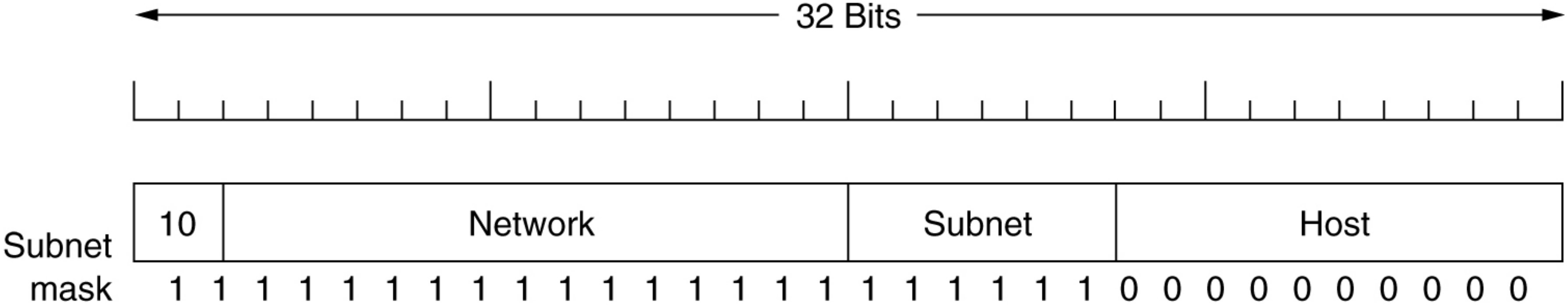
- Address classes



Class	32 Bits		Range of host addresses
A	0 Network	Host	1.0.0.0 to 127.255.255.255
B	10 Network	Host	128.0.0.0 to 191.255.255.255
C	110 Network	Host	192.0.0.0 to 223.255.255.255
D	1110	Multicast address	224.0.0.0 to 239.255.255.255
E	1111	Reserved for future use	240.0.0.0 to 255.255.255.255

IP address: more

- Problem with “address classes”
  - too big a Class A network
  - too (many) small Class C networks
- Classless address
  - CIDR: classless inter-domain routing



32 Bits

	10	Network	Subnet	Host
Subnet mask	1 1	1 1 1 1 1 1 1 1 1 1 1 1 1 1	1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0

UVic's IP space

- UVicNet
  - Class B: 142.104.0.0/16
- UVic EngNet
  - network address: 142.104.96.0
  - network mask: 255.255.224.0
    - 142.104.96.0/19
  - subnet test
    - net_add & net_mask ?= host_add & net_mask
    - host_A_add & net_mask ?= host_B_add & net_mask

Obtain an IP address

- Static configuration
  - e.g., on UVic campus
  - e.g., /etc/sysconfig/network-scripts/ifcfg-eth0
- Allocated by service provider
  - e.g., at home
  - DHCP: dynamic host configuration protocol
    - obtain IP add, net mask, default gateway, DNS, etc
    - authentication often needed

Make linux box act as router

1) **A/B/C**: sysctl -w net.ipv4.ip-forward=1

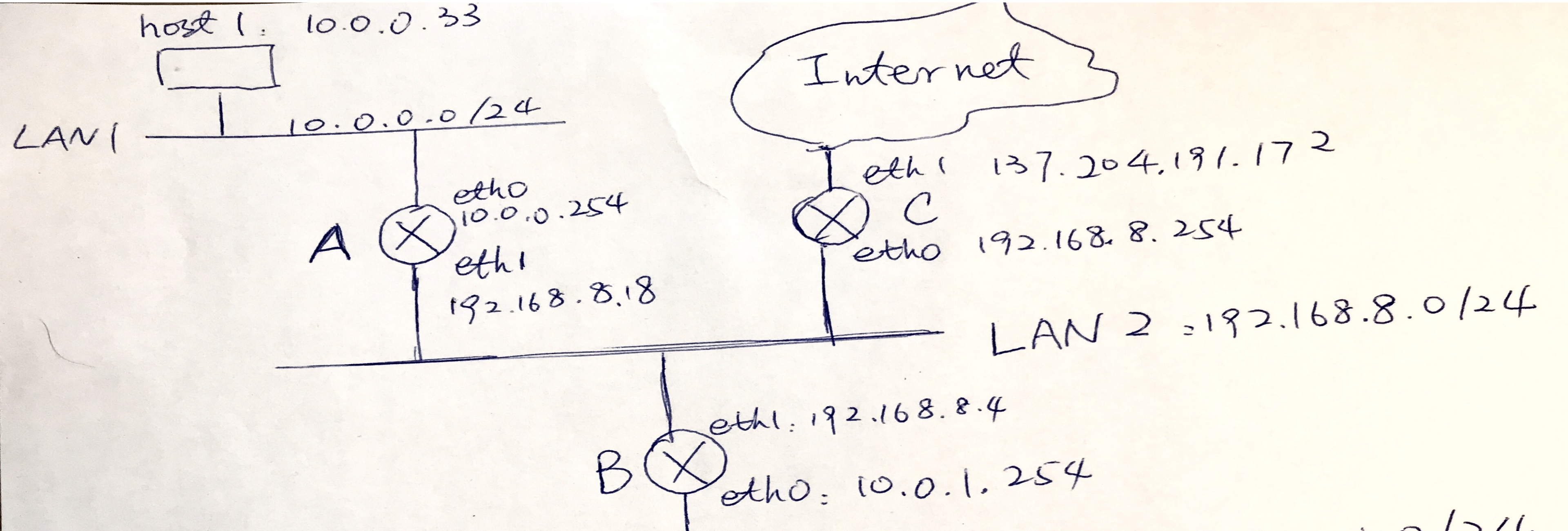
2) **A/B/C**: ifconfig eth1 *.*.*.* netmask 255.255.255.0 broadcast *.*.*.255

3) at **A**: route add default gw 192.168.8.254 dev eth1

```
route add -net 10.0.1.0/24 gw 192.168.8.4 dev eth1
```

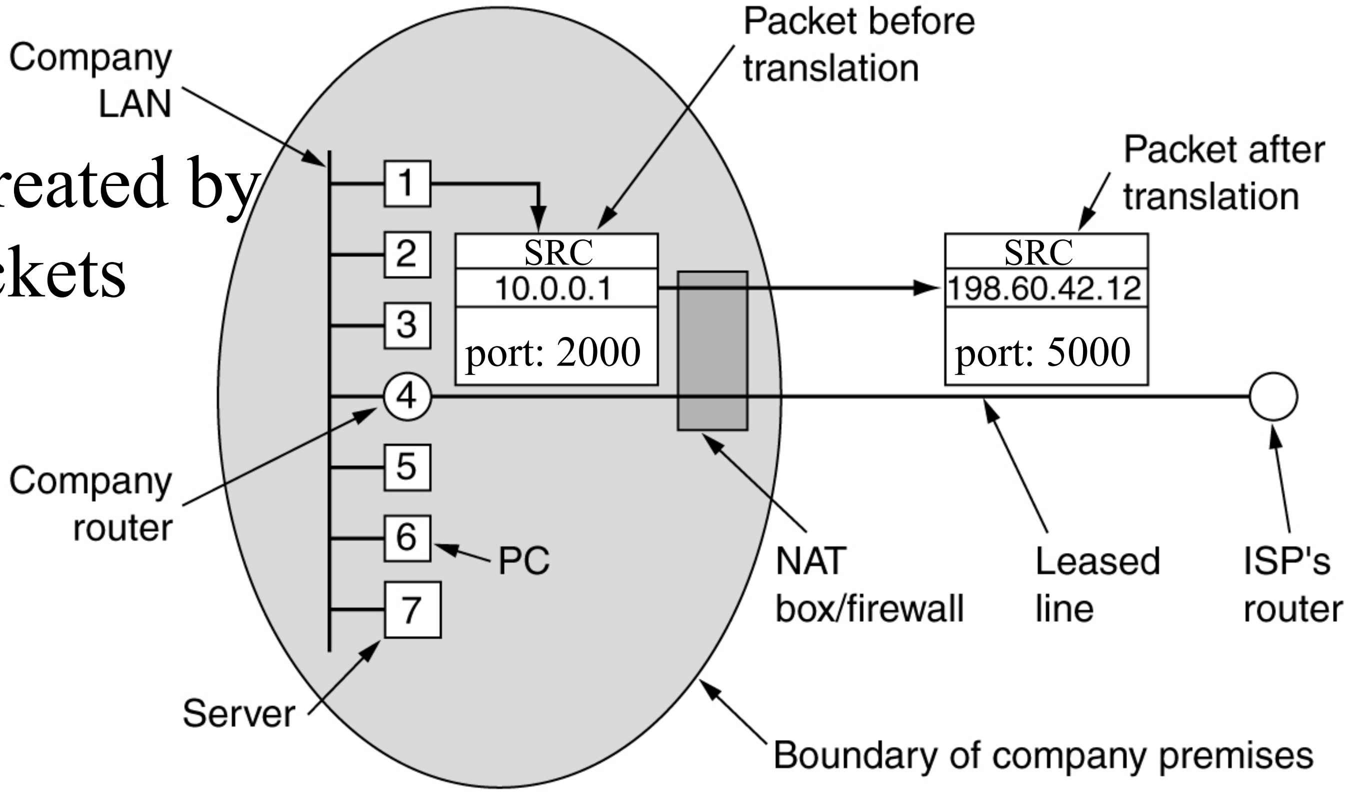
at **C**: route add -net 10.0.0.0/24 gw 192.168.8.18

```
route add -net 10.0.1.0/24 gw 192.168.8.4
```



Network address translation

- "Not enough IP addresses!"
  - not efficiently allocated, more connected devices, etc
- NAT/NAPT
  - translation created by outgoing packets
- Problems
  - NAT solves
  - NAT creates



Header fields

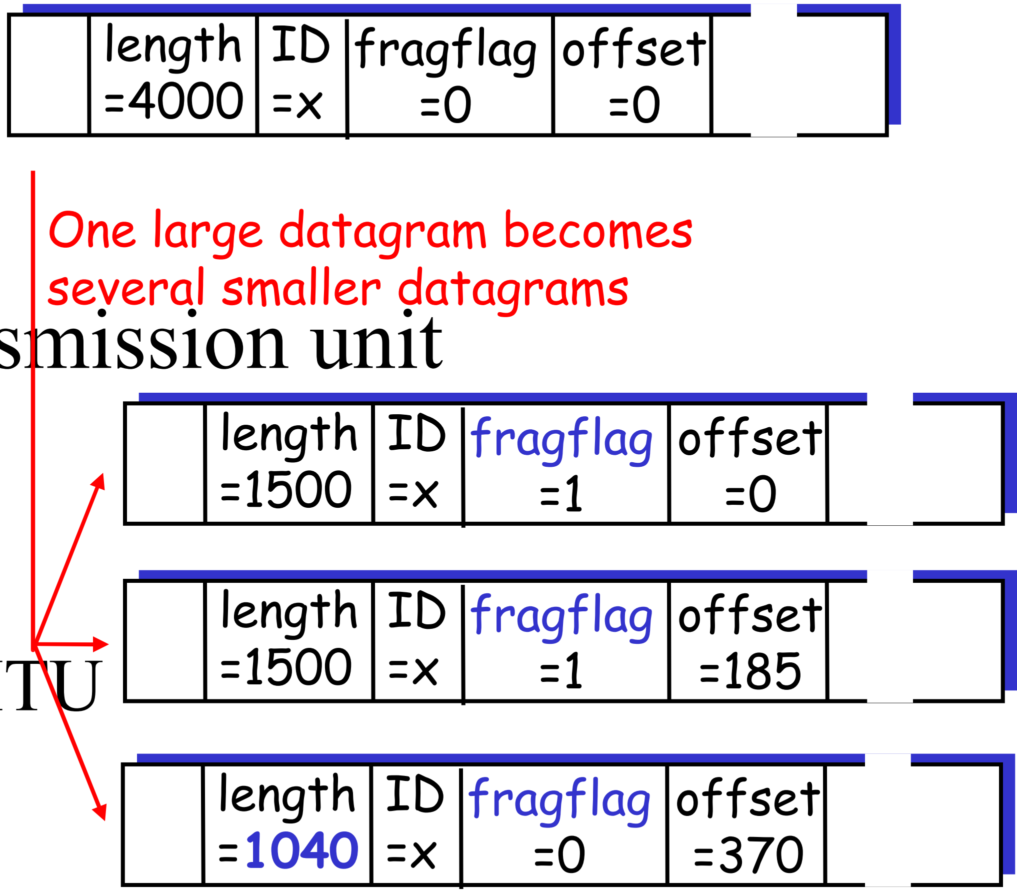
- IP header checksum (16-bit)
  - TCP/IP-style checksum
  - cover IP header (and option) only
- Protocol ID (8-bit)
  - TCP(6), UDP(17); /etc/protocols
- TTL: time-to-live (8-bit)
  - decrement by each router
  - drop if TTL=0

Header field: more

- Total length (16-bit)
  - byte counter
- IHL: IP header length (4-bit)
  - 4-byte counter
- Identification (16-bit)
- Fragment offset (13-bit)
  - 8-byte offset
  - DF: don’t fragment; MF: more fragment(s)

Fragment and reassemble

- IP packet length
  - 2^{16}-1 bytes
- MTU: maximum transmission unit
  - Ethernet: 1500 bytes
- Fragment
  - when total length > MTU
- Reassemble
  - only at destination
- PMTU discovery



length	ID	fragflag	offset
=4000	=x	=0	=0

One large datagram becomes several smaller datagrams

length	ID	fragflag	offset
=1500	=x	=1	=0

length	ID	fragflag	offset
=1500	=x	=1	=185

length	ID	fragflag	offset
=1040	=x	=0	=370

Type of service

- ToS: type of service
  - precedence (bit 7-5)
    - 0: normal traffic
    - 7: network control traffic
  - Flags (bit 2-4): Delay, Throughput, Reliability
- New definition: DiffServ Code Point
  - per-hop behavior (bit 7-2)
- The other two bits

Summary

- IP
  - IP addressing
    - address class, classless, NAT
  - fragmentation and reassembly
    - MTU, “total length”, offset
- Explore further
  - /sbin/ifconfig

Next

- Routing algorithm