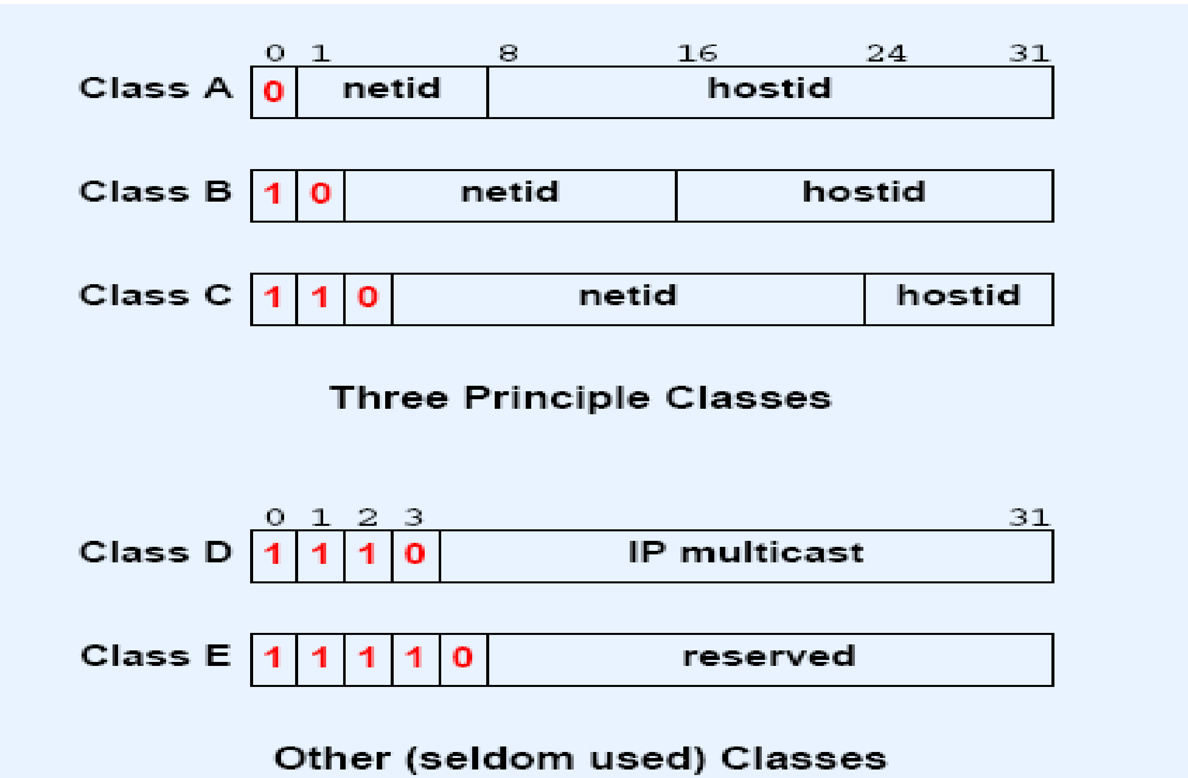
Introduction to Computer Networks & Communications

Lecture 7-8: Network Layer

Dr. Amal ElNahas

Original IPv4 Address Classes



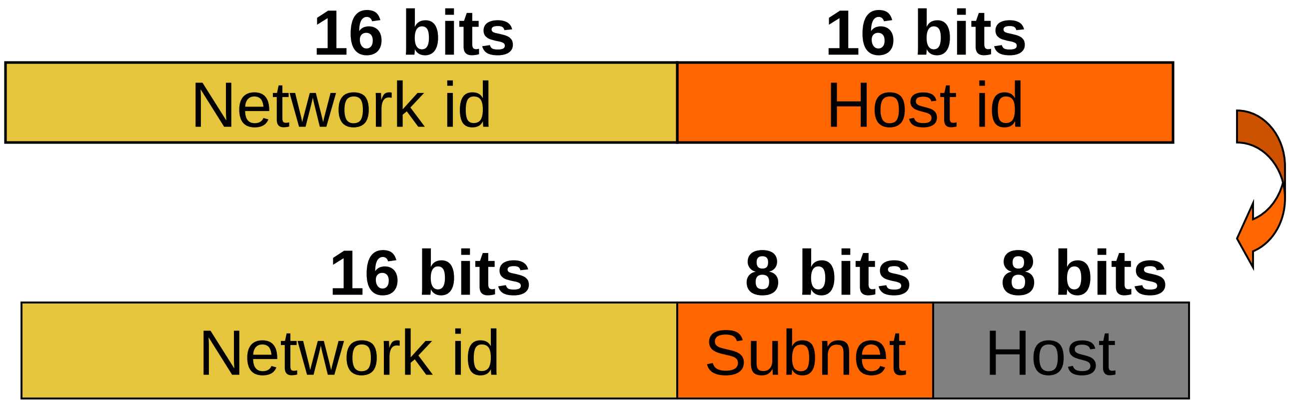
class				
A	NetID	hostID	hostID	hostID
B	NetID	NetID	hostID	hostID
C	NetID	NetID	NetID	hostID

Classful addressing: Good or Bad?

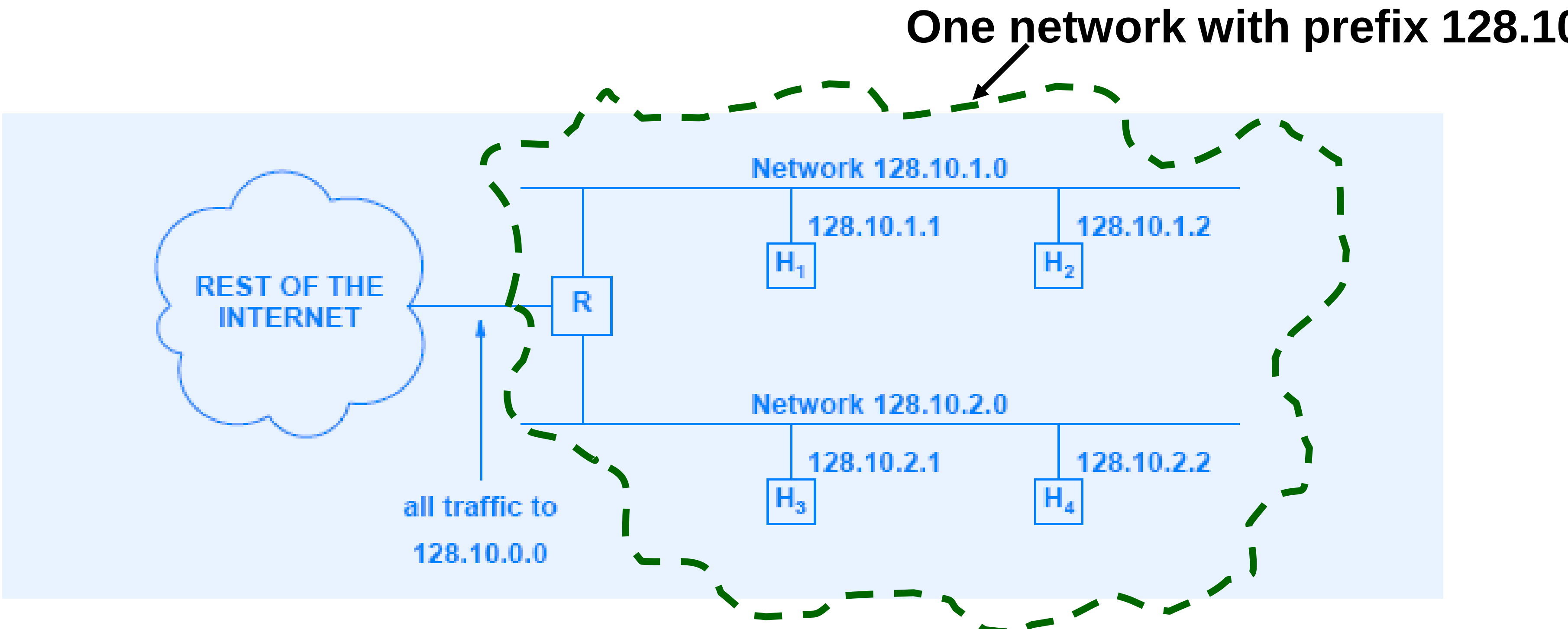
- Good: simple, easy to understand
- Bad: limited address space
  2^32 = 4G addresses, not enough?
  IPv6
- Bad: limited network size choices (3)
  Ex.: what if a class C net needs to grow beyond 255 hosts?
  Subnetting
- Bad: moving to a new network requires changing IP addresses
  Mobile-IP

Subnetting

- A class B address is divided into two parts: **network** part and **local** part
- Local part is further divided locally into **subnet** and **host** parts
- Splitting is done internally, yet looks like a single network to the outside world



Example



- 2 physical networks sharing the same network prefix 128.10 (same organization)
- Router R uses third byte to differentiate between the 2 networks
- Appears as a single network with prefix 128.10 for the outside world

Subnet Mask

- Subnet mask needed to differentiate between different subnets
- Allows hosts to determine if another IP address is on the same subnet or the same network

	16 bits	8 bits	8 bits
	Network id	Subnet id	Host id
Mask	1111111111111111	11111111	00000000
	255.255	.255	.0

- 1’s represent network part, 0’s represent host part

Subnet mask: Example 1

Assume an organization with multiple subnets is assigned address 150.100.0.0. Assume each subnet has up to 120 hosts

- How many host bits do we need?
- What is the maximum number of subnets
- What is the network mask?

Take Home assignment

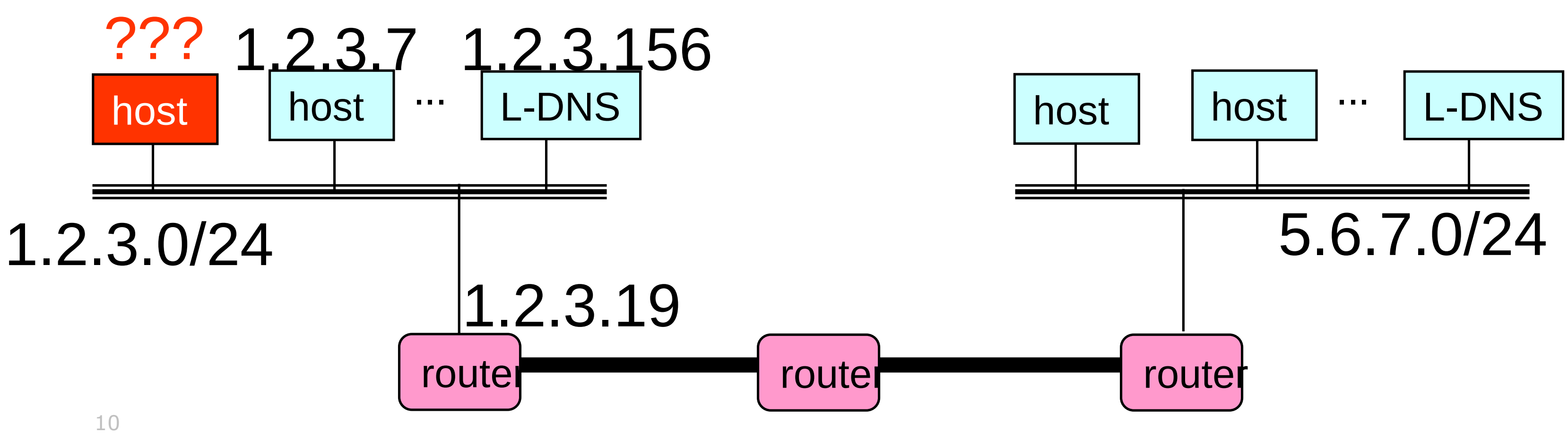
Consider a class B network 166.113.0.0, with a total of 15 subnets and the largest has 450 hosts. Suggests four acceptable options for subnetting. Which one would you choose?

CIDR: Classless InterDomain Routing

- Do not use classes to determine network ID. Network part can be any number of bits long
- Address written as a.b.c.d/x
  - a.b.c.d: IP prefix
  - x: address mask length (how many bits used to specify the network id)
- Example:
  214.5.480.0/20
  - Prefix occupies 20 bits
  - Suffix occupies 12 bits
- Class A network is a /8
- Class B network is a /16
- Class C network is a /24

How To Get an IP address?

- What IP address the host should use?
- How to send packets to remote destinations?
- How to ensure incoming packets arrive?



How to Get an IP Address??

- All hosts on same network assigned same address prefix
  - Prefixes assigned by central authority
  - Obtained from ISP

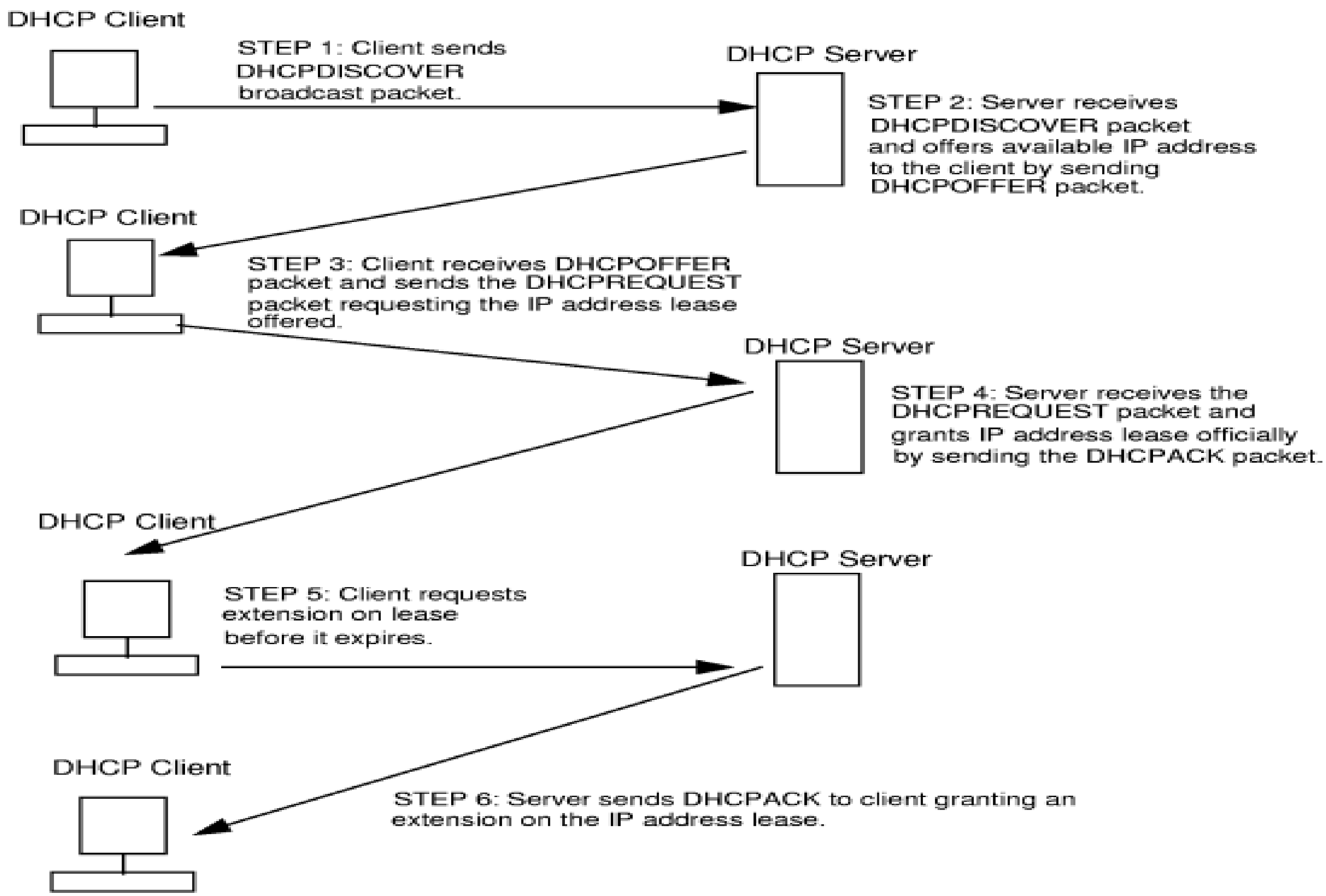
- Each host on a network has a unique suffix
  - Assigned locally by system admin
  - Local administrator must ensure uniqueness

- **DHCP**: **D**ynamic **H**ost **C**onfiguration **P**rotocol: dynamically get address from a server

Dynamic Host Configuration Protocol (DHCP)

- Assigns IP addresses to hosts dynamically
- Allows host to learn its own network parameters
- On startup, host broadcasts DHCP discovery message
  - Destination IP address: 255.255.255.255
  - Source IP address: 0.0.0.0 (this host)
- DHCP server responds
  - Grants lease for an IP address, network mask, lease time
  - Host must renew lease or stop using address when lease expires

DHCP client-server Interaction

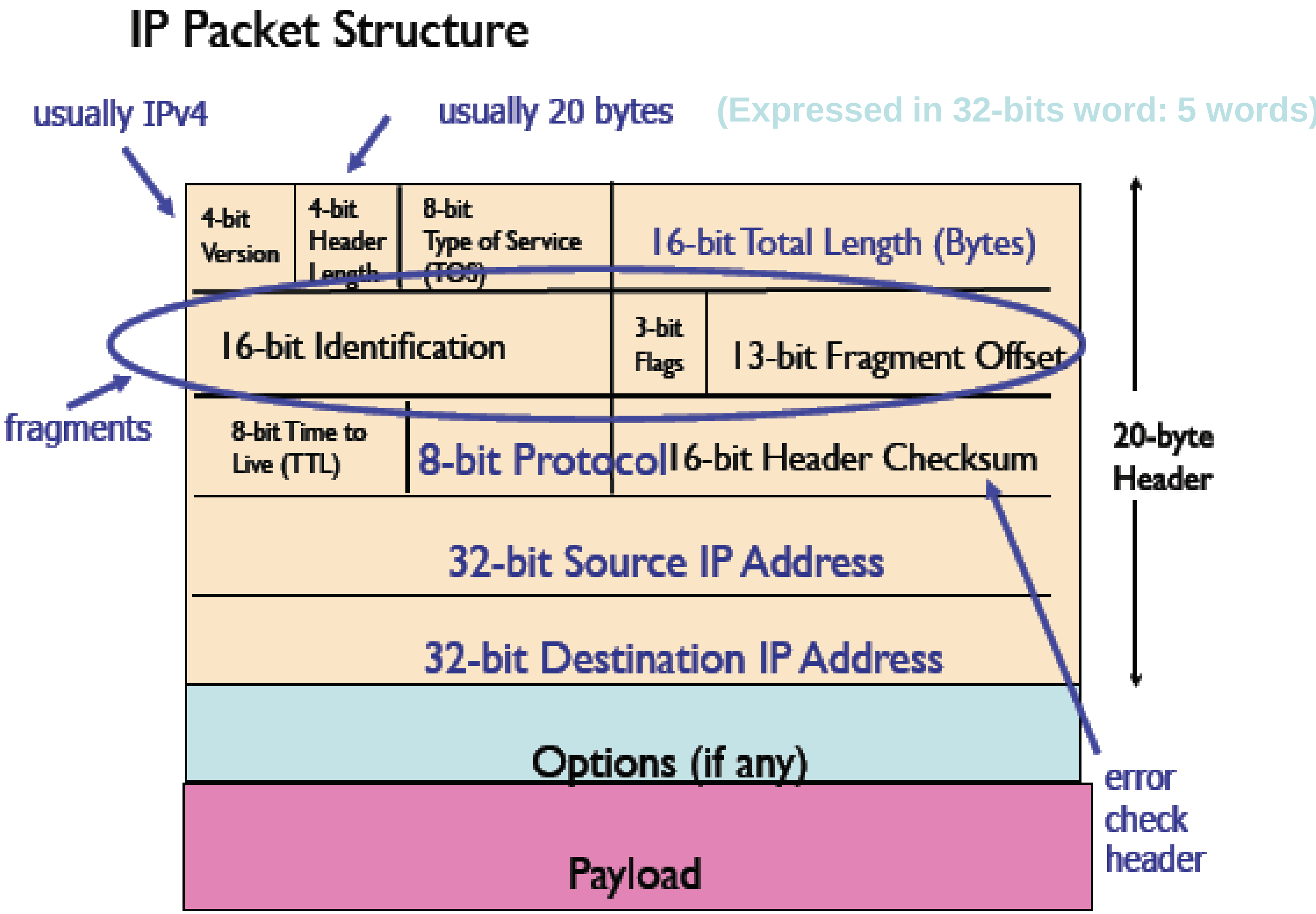


IP Datagram Format

Header	**Data Area**

- Each IP datagram is composed of 2 parts:
  - Datagram header: minimum 20 bytes (5 words) and maximum 60 bytes (15 words)
  - Datagram data
- Maximum size of a datagram (including the header) is 65,535 bytes (actual size much smaller)

IP datagram Format



IP datagram Format: Details

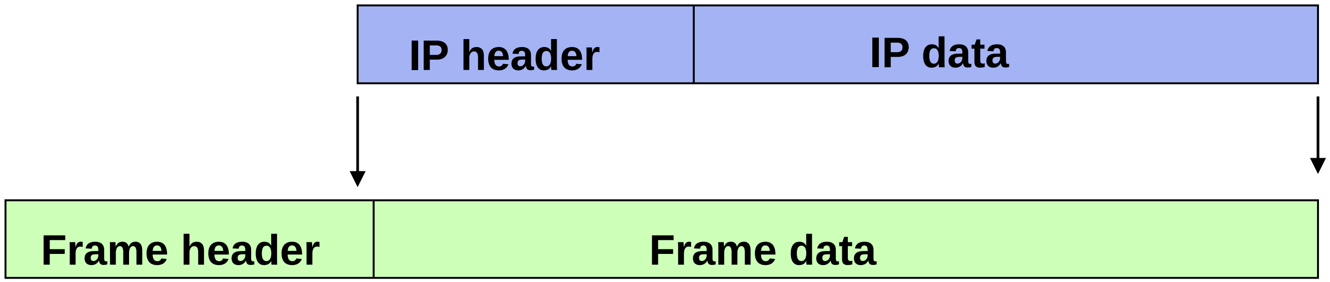
- Version (4 bits): currently IPv4
- header length (4 bits): in 32-bit words, 5 if no options
- type of service (8 bits): datagram’s precedence, desired characteristics (low delay,..). Late 90’s: DiffServ
- total length (16 bits): in bytes (2^16 – 1 bytes), header + data
- datagram identifier: allows destination to match up fragments of the same datagram
- flags
  - more-fragments: says this isn’t the last fragment of the datagram
  - don’t-fragment: prohibits fragmentation; packet will be dropped rather than fragmented

IP datagram Format: Details

- Offset (13 bits): offset within datagram at which this fragment begins (given in multiple of 8 bytes)
- time to live: initially set to 64(or higher); decremented on each hop; packet dropped if TTL==0
- protocol: identifies which higher-level protocol this datagram belongs to
- checksum: 16-bit ones-complement sum (over header only, recomputed at each router)
- source address, destination address: obvious
- options: rarely used (timestamp, routers to visit,..), starts with an option code octet, padding needed (header is multiple of 32)

Datagram Encapsulation

- Network hardware treats datagram as data
- Datagram is encapsulated by adding the appropriate header forming a frame



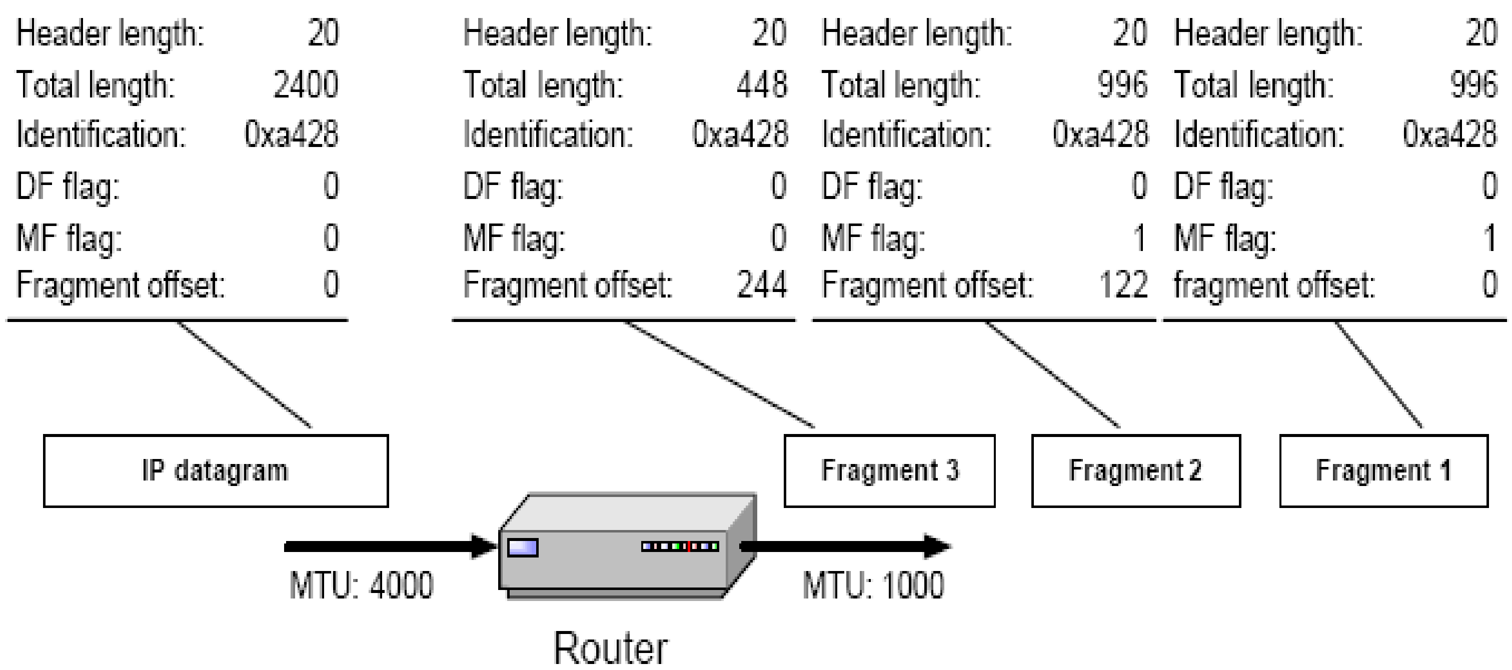
- Potential problem:
  - Each network has a maximum transmission unit (MTU); the maximum allowable frame size it can handle (ex: Ethernet 1500,..)
  - Packet may travel through diverse networks with different MTUs
  - A datagram size can exceed MTU

Solution

- If packet is bigger than MTU, break it into fragments (fragmentation)
- Send each piece in a frame
- Reassemble at ultimate destination (reassembly)
- IP header fields used to identify and order related fragments

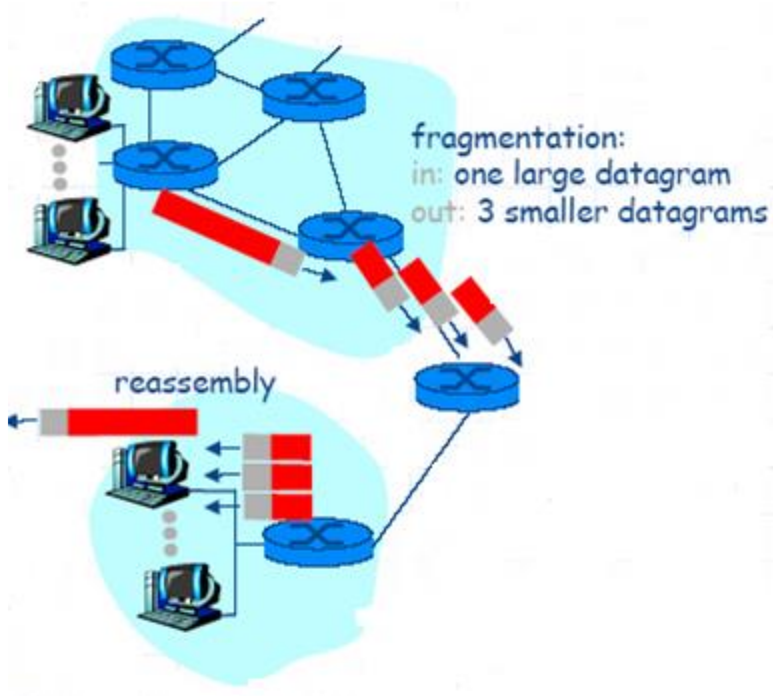
Example

- Initial datagram length = 2400 bytes
- Net1 MTU = 4000, Net2 MTU = 1000
- Size of fragments 1 and 2: 996 (976+20); why 976???



Reassembly

- Performed by destination host
- Store fragments in memory until they all show up
    - Timer is used to ensure all fragments arrive (value between 60 sec and 120 sec):
    - Timer starts when first fragment arrives
    - If timer expires, discard the whole diagram



To be continued….