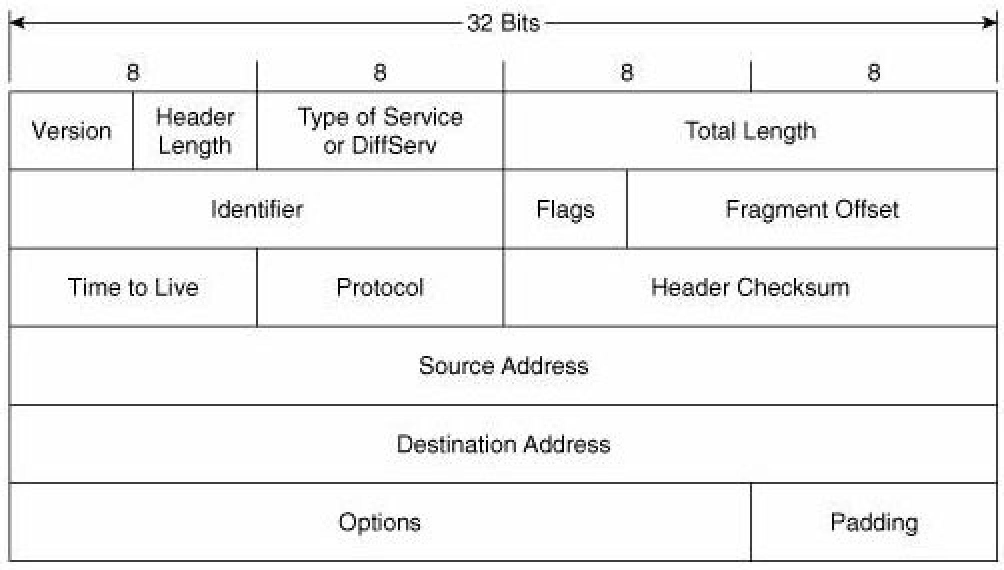
IPv4 addressing & subnetting

+ fragmentation

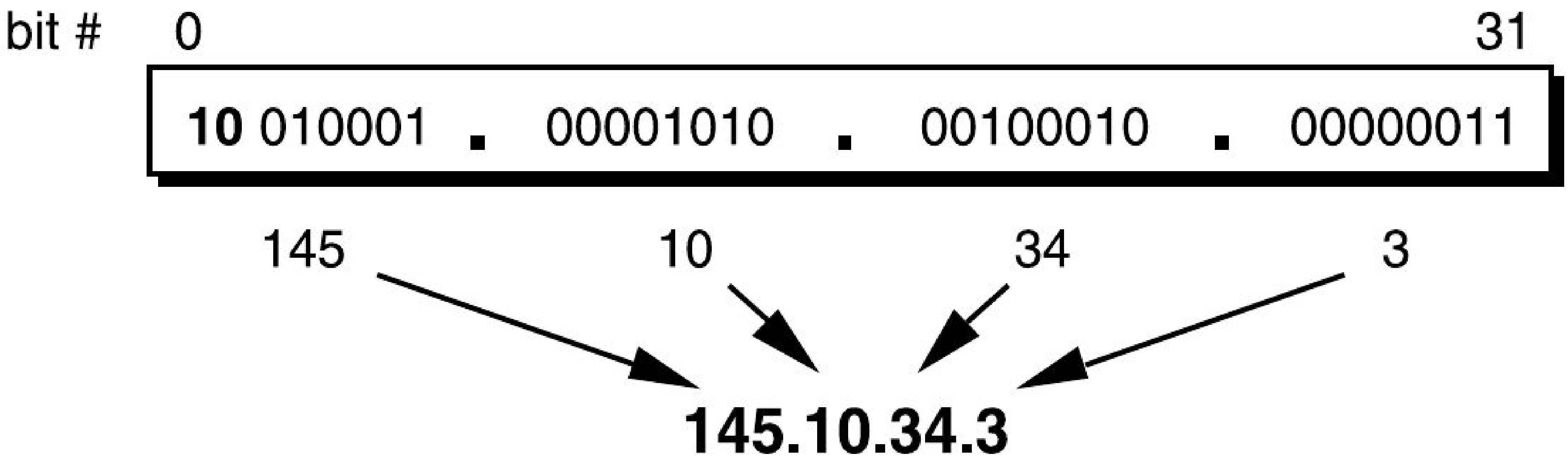
CS335a - Introduction to Computer Networks

Kostis Triantafyllakis - ctriant@csd.uoc.gr

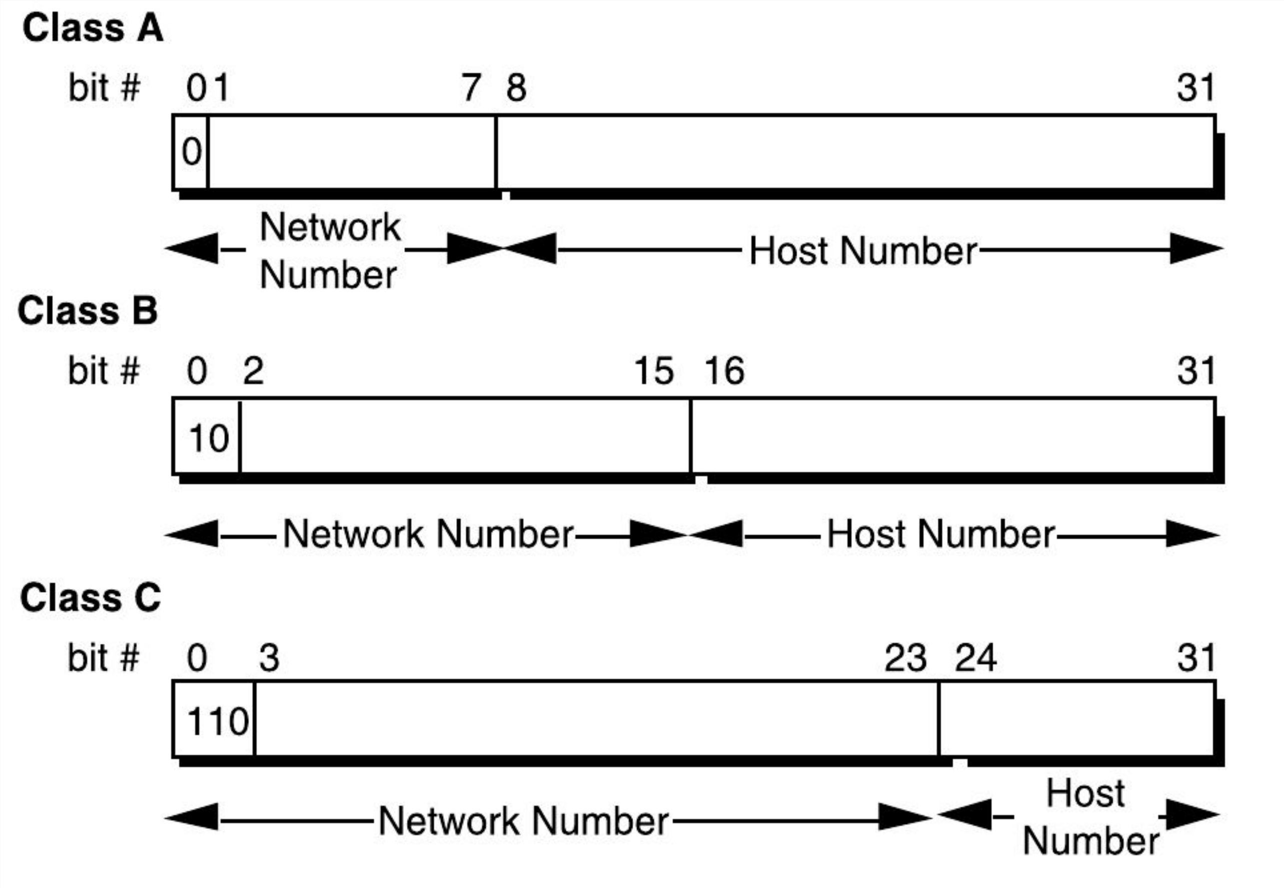
IPv4 header


32 Bits
8
8
8
8
Version
Header Length
Type of Service or DiffServ
Total Length
Identifier
Flags
Fragment Offset
Time to Live
Protocol
Header Checksum
Source Address
Destination Address
Options
Padding

Dotted-Decimal notation



Classful IP addressing



126 (2^7-2) networks
~16,000,000 hosts

~16k (2^{14}) networks
~65k (2^{16}-2)hosts

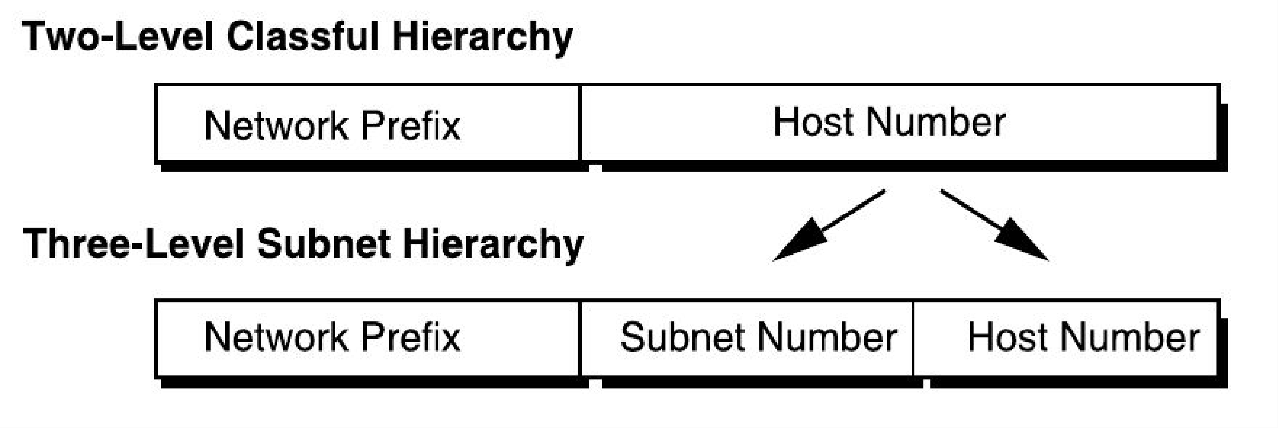
~2M (2^{21}) networks
~254 (2^8-2)hosts

Unforeseen limitations to Classful Addressing

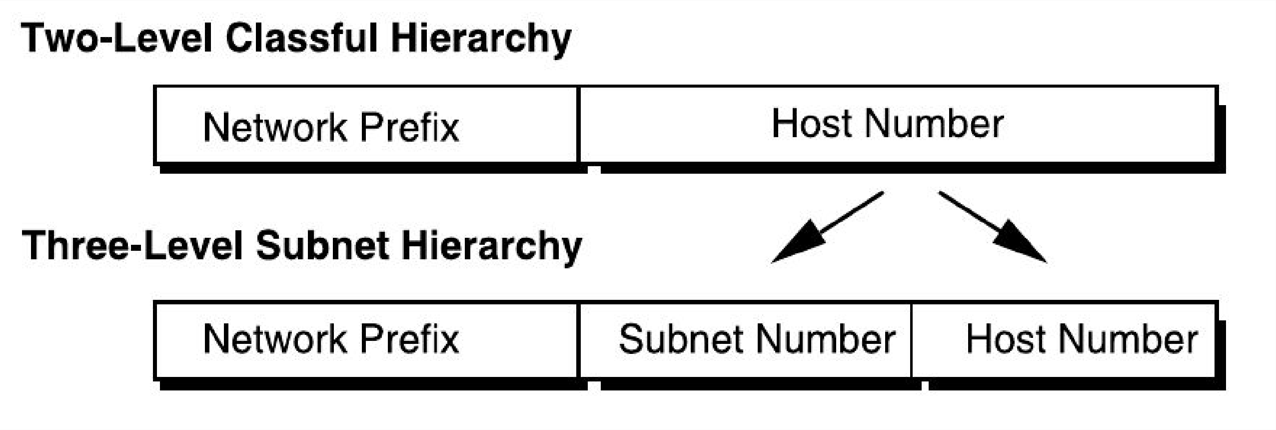
- The original designers never envisioned the current Internet growth
- Addresses were freely assigned to those who asked for them without concerns.
- The decision of 32-bit addresses was wrong.
  4,294,967,296 addresses are not enough
- The concept of the Classful Addressing was easy to understand and implement, but it was not efficient for a finite address space.
  - /24 supports 254 hosts that is small
  - /16 supports 65,534 hosts that is big

Subnetting

- The division of a single Class A, B or C network into smaller pieces
- What need led to Subnetting?
  - Internet routing tables were beginning to grow
  - Local admins had to request another network number from the Internet before a new network could be installed in their site.

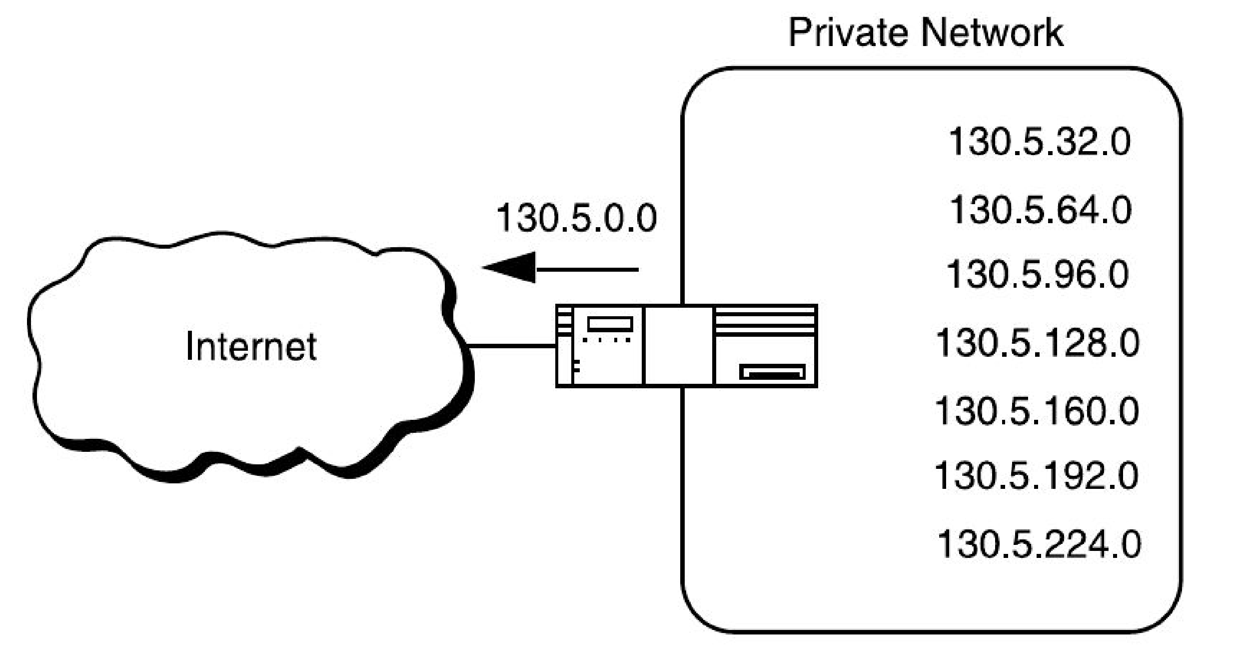


Subnetting



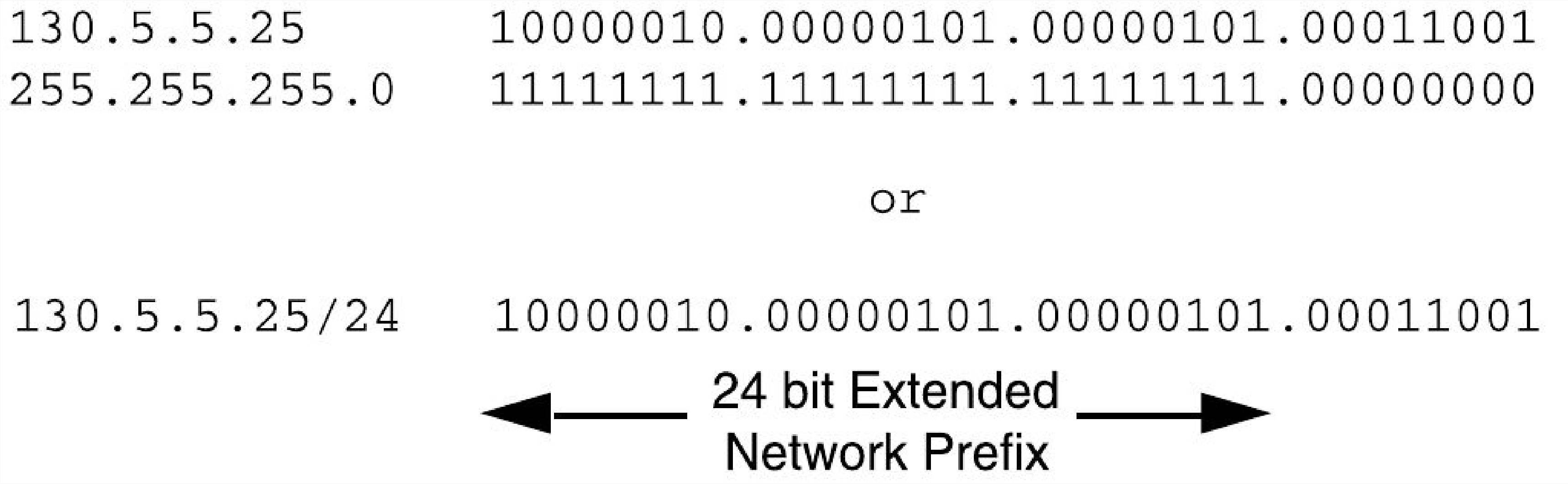
- Subnet structure of a network is never visible outside the organization's private network
- Each organization is assigned one (or at most a few) network addresses from the IPv4 address space.
  - The organization was free to assign a distinct sub-network number to each of its internal networks.

Subnetting



- The size of Internet routing table is not affected
- Rapid changing of routes within the private network do not affect the Internet routing table

Subnet mask



- Internet routers use only the Network prefix of the destination address to route the traffic to a subnet.
- Routers within a subnet use the Extended Network Prefix to route the traffic between the individual subnets.

Subnet design considerations

- What is the total number of subnets that are needed today?
- What is the total number of hosts that are needed today?

- What about the future?

Example

Define the subnet and host addresses

- An organization holds the network number: **193.1.1.0 / 24**
- Needs to define **6 subnets**
- The largest subnet is required to support **25 hosts**

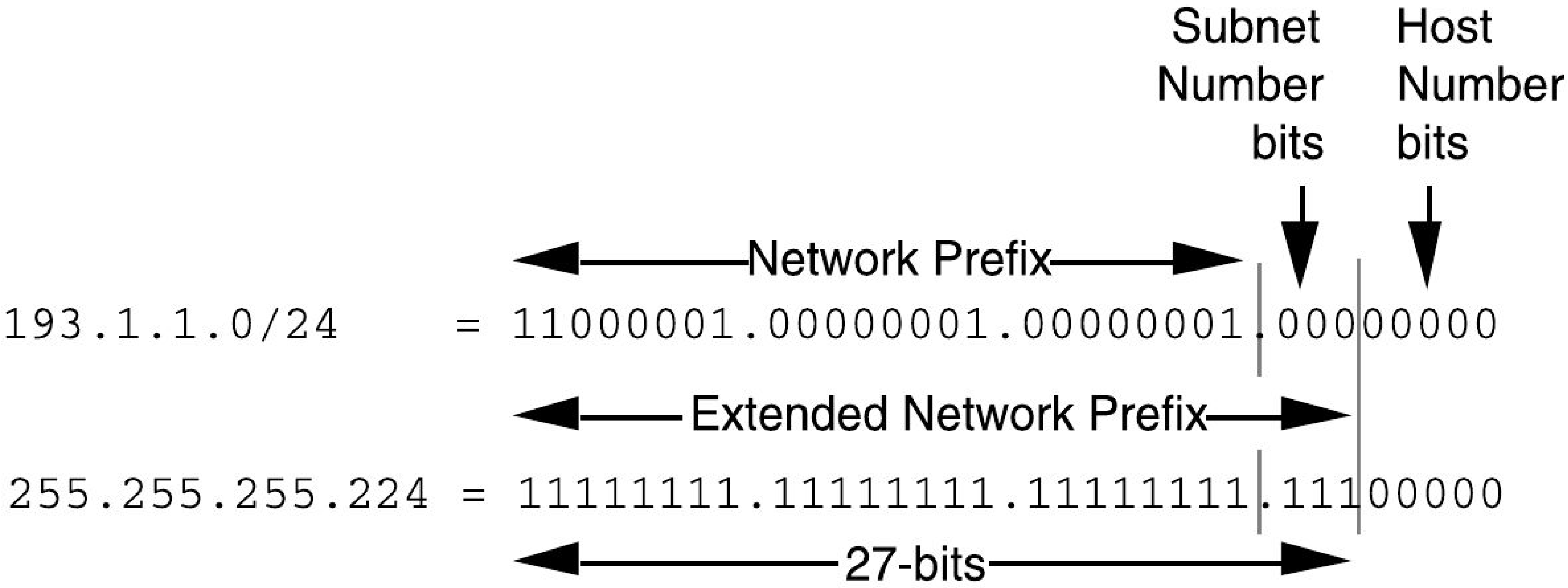
Example

Decimal: 193.1.1.0/24
Binary: 11000001.00000001.00000001.00000000

- What is the number of bits required to define 6 subnets?
  - 3bits because $2^3 = 8$ subnets
  - That leaves 2 spare subnets.

- What's the subnet mask?
  - Since the organization is subnetting a /24 it needs 3 more bits into the mask or equivalently /27

Example


Subnet Number bits
Host Number bits
Network Prefix
193.1.1.0/24 = 11000001.00000001.00000001.00000000
Extended Network Prefix
255.255.255.224 = 11111111.11111111.11111111.11100000
27-bits

Example

Base Net: 11000001.00000001.00000001 .00000000 = 193.1.1.0/24
Subnet #0: 11000001.00000001.00000001.**000** 00000 = 193.1.1.0/27
Subnet #1: 11000001.00000001.00000001.**001** 00000 = 193.1.1.32/27
Subnet #2: 11000001.00000001.00000001.**010** 00000 = 193.1.1.64/27
Subnet #3: 11000001.00000001.00000001.**011** 00000 = 193.1.1.96/27
Subnet #4: 11000001.00000001.00000001.**100** 00000 = 193.1.1.128/27
Subnet #5: 11000001.00000001.00000001.**101** 00000 = 193.1.1.160/27
Subnet #6: 11000001.00000001.00000001.**110** 00000 = 193.1.1.192/27
Subnet #7: 11000001.00000001.00000001.**111** 00000 = 193.1.1.224/27

Example

Subnet #2: 11000001.00000001.00000001.010 **00000** = 193.1.1.64/27

Host #1: 11000001.00000001.00000001.010 **00001** = 193.1.1.65/27

Host #2: 11000001.00000001.00000001.010 **00010** = 193.1.1.66/27

Host #3: 11000001.00000001.00000001.010 **00011** = 193.1.1.67/27

Host #4: 11000001.00000001.00000001.010 **00100** = 193.1.1.68/27

Host #5: 11000001.00000001.00000001.010 **00101** = 193.1.1.69/27

.

.

Host #15: 11000001.00000001.00000001.010 **01111** = 193.1.1.79/27

Host #16: 11000001.00000001.00000001.010 **10000** = 193.1.1.80/27

.

.

Host #27: 11000001.00000001.00000001.010 **11011** = 193.1.1.91/27

Host #28: 11000001.00000001.00000001.010 **11100** = 193.1.1.92/27

Host #29: 11000001.00000001.00000001.010 **11101** = 193.1.1.93/27

Host #30: 11000001.00000001.00000001.010 **11110** = 193.1.1.94/27

IP fragmentation

- **Maximum Transmission Unit** (MTU) defines **the largest packet size** that can traverse this path **without suffering fragmentation**
- **If an IP datagram has size larger than the MTU, then it is fragmented into smaller pieces before it is sent.**

Example: Suppose we want to transmit an IP datagram of size 3000 bytes through a link of MTU 500 bytes. How many fragments are produced and what are the values of the offset field in each of the headers?

Example

Example: Suppose we want to transmit an IP datagram of size 3000 bytes through a link of MTU 500 bytes. How many fragments are produced and what are the values of the offset field in each of the headers?

IP fragment payload = 500 bytes (MTU) - 20 bytes (min IPv4 header) = 480 bytes

IP datagram of interest payload = 3000 - 20 = 2980 bytes

Total # of segments = IP datagram of interest payload / IP fragment payload
= 2980 / 480
= 6.2
= 7 (The last packet will have smaller payload than the available 480 bytes)

Example

Example: Suppose we want to transmit an IP datagram of size 3000 bytes through a link of MTU 500 bytes. How many fragments are produced and what are the values of the offset field in each of the headers?

What about the header fields?

Segment 0:	0 - 479 bytes of original	offset = 0	more = 1
Segment 1:	480 - 959	offset = 60	more = 1
Segment 2:	960 - 1439	offset = 120	more = 1
Segment 3:	1440 - 1919	offset = 180	more = 1
Segment 4:	1920 - 2399	offset = 240	more = 1
Segment 5:	2400 - 2879	offset = 300	more = 1
Segment 6:	2880 - 2980	offset = 360	more = 0