



# SLIIT

*Discover Your Future*

# IT2050 - Computer Networks

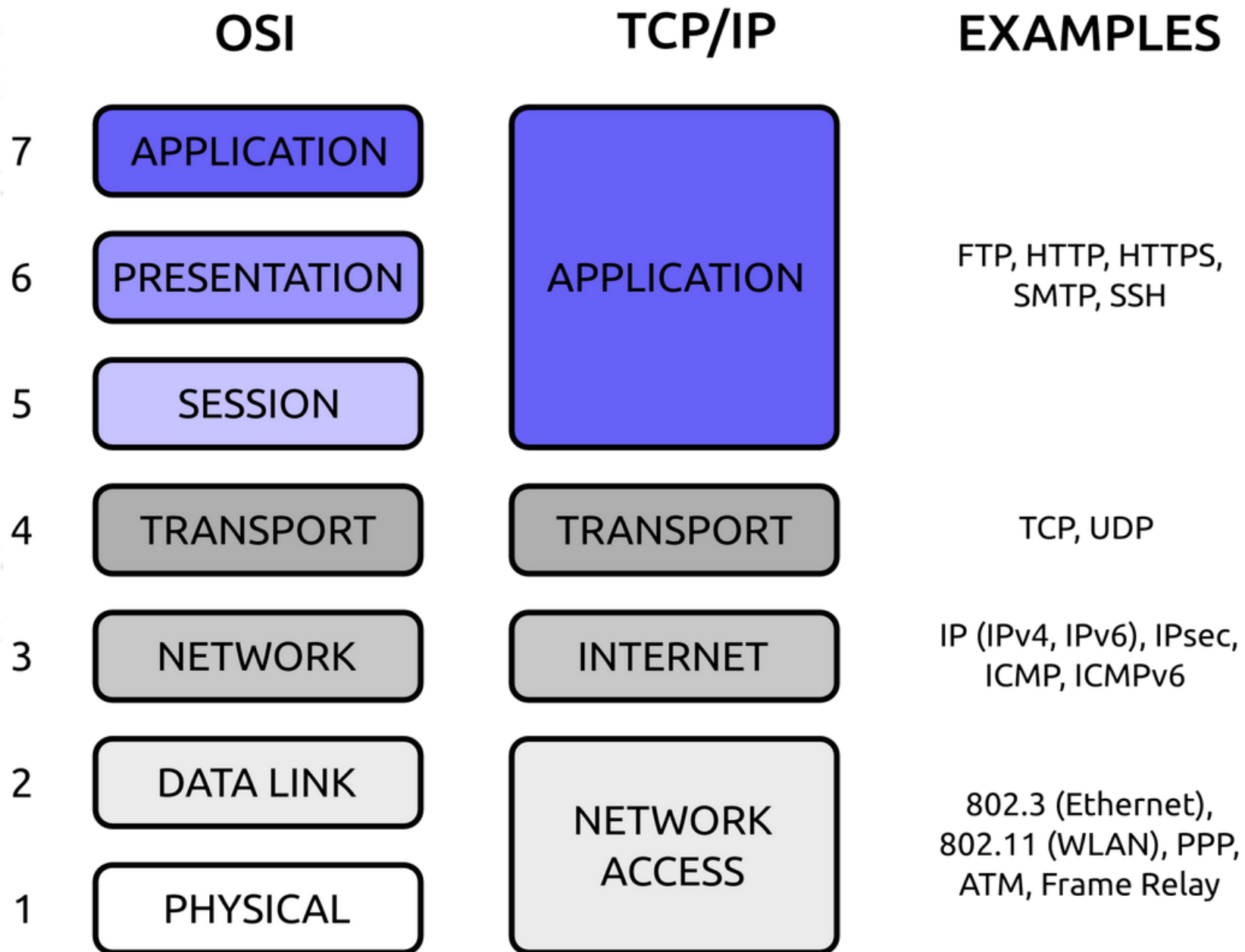
**Lecture 5**

**Internet Protocol**

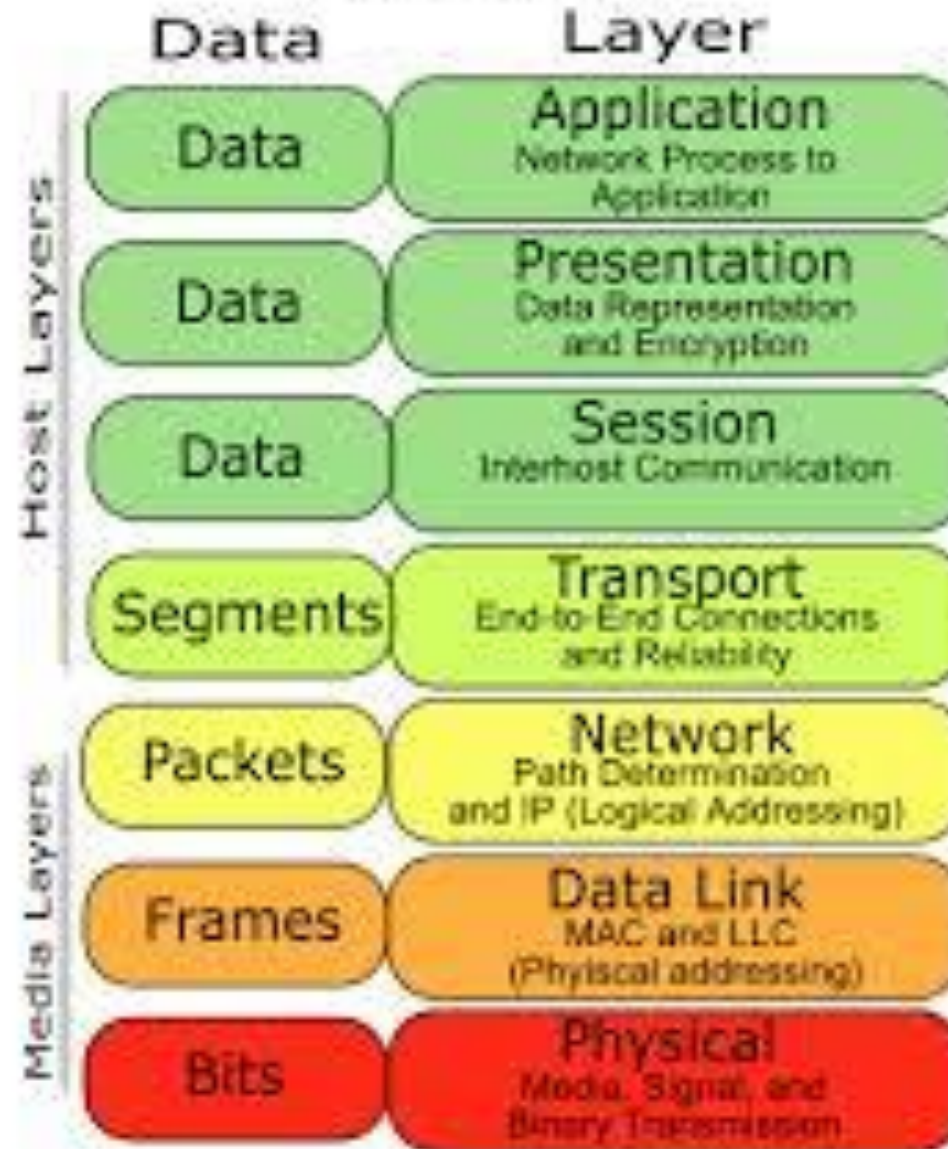
**Ms. Hansika Mahaadikara**

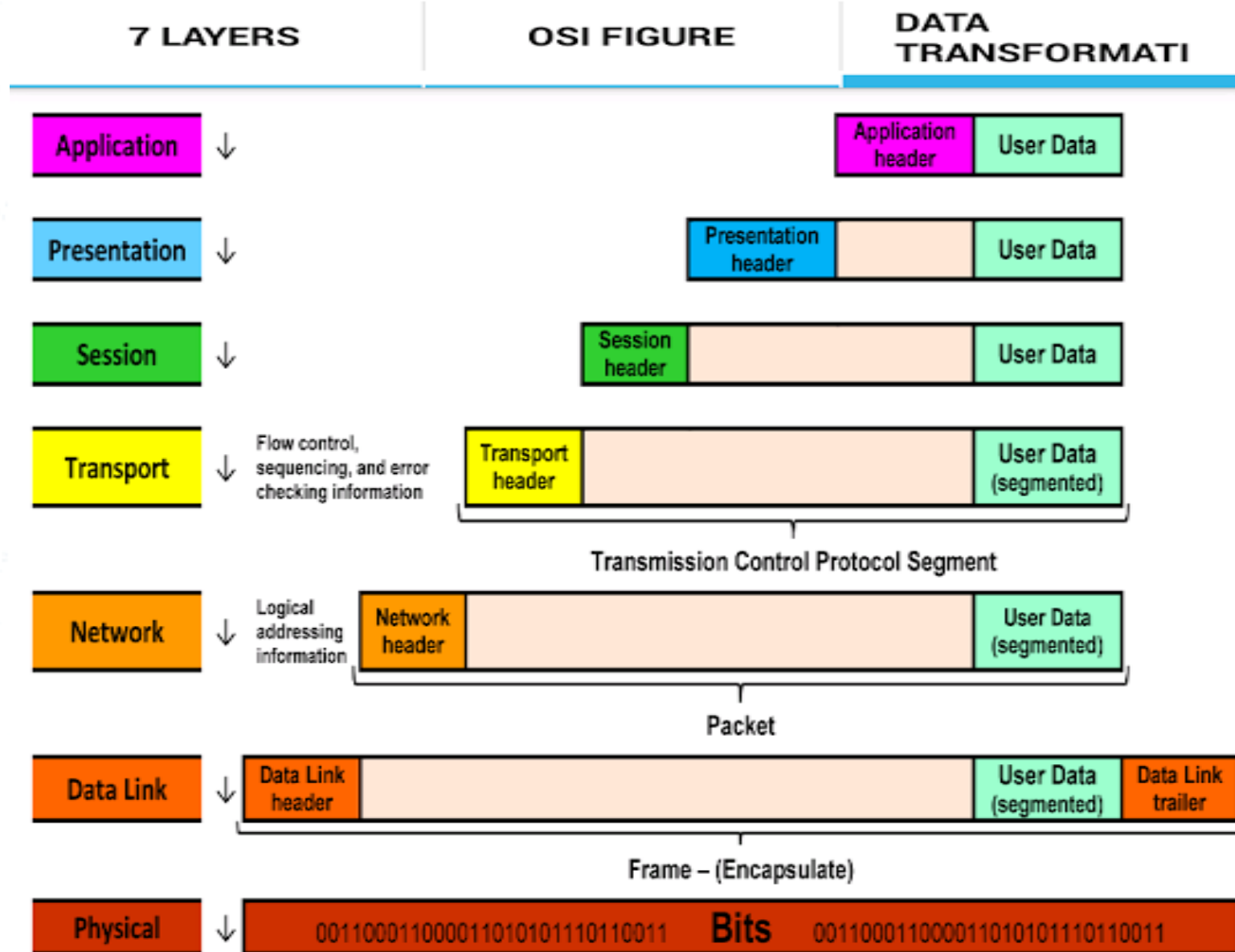


**SLIIT**  
**FACULTY OF COMPUTING**



# OSI Model





Format of the information at each layer

# Overview of IP

- IP is the network layer protocol of TCP/IP
- No Error control, flow control and congestion control

Hence IP is an unreliable protocol

- Combination TCP/IP is reliable

But UDP/IP is an unreliable combination

- IP packets operate as datagram
- IP packets originated from same source can travel through different routes and reach the destination at different times
- Therefore IP packets may reach the destination out of order

# Identification

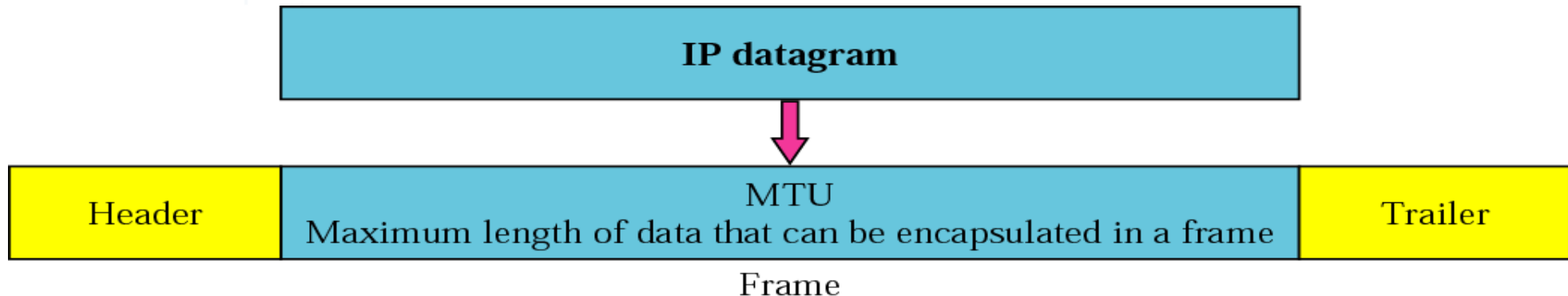
- Each IP packet is identified by a serial number called “Identification”
- This sequence will be helpful to the receiver to reassemble the packets in the correct order, although they may receive in out of order



# Maximum Transmission Unit (MTU)

- Maximum amount of data that can be accommodated in a frame

| <i>Protocol</i>      | <i>MTU</i> |
|----------------------|------------|
| Hyperchannel         | 65,535     |
| Token Ring (16 Mbps) | 17,914     |
| Token Ring (4 Mbps)  | 4,464      |
| FDDI                 | 4,352      |
| Ethernet             | 1,500      |
| X.25                 | 576        |
| PPP                  | 296        |



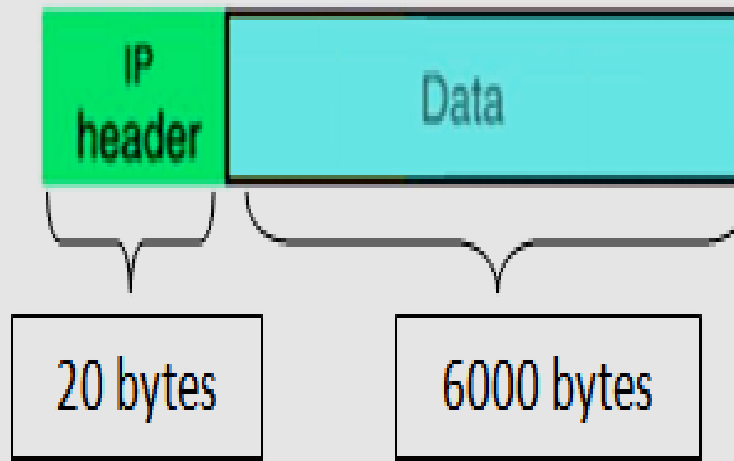
# Fragmentation

- If the IP packet size is bigger than MTU, it should be fragmented

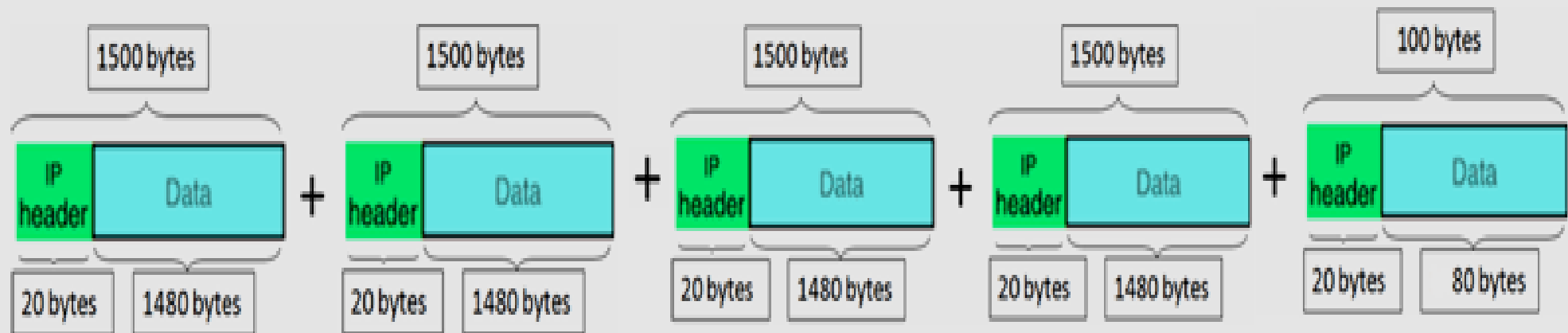
Ex : If the original packet has 6000 bytes of data

1. Separate data and Header of IP Packet
2. Break data part into MTUs (Fragments)
3. Add 20 byte header to each fragment





Fragmentation



# Fragmentation offset

- The identification of each fragmented IP packet is equal to identification of original IP packet

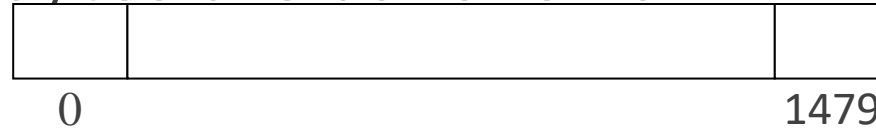
Ex : If the identification of original IP packet is 2000,  
identification of all five fragments is 2000

- “ **Fragmentation offset** ” is an another parameter used to identity the order of fragments

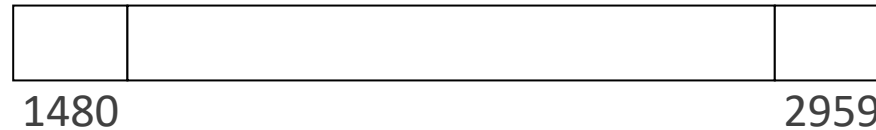
# Fragmentation offset cont.

- If the original packet has 6000 bytes of data, the numbering of data bytes are as follows

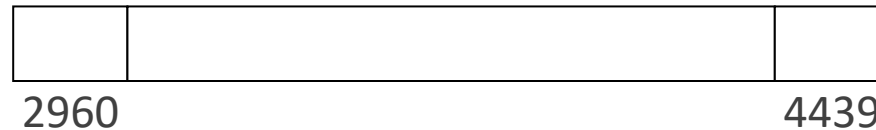
Fragment 1



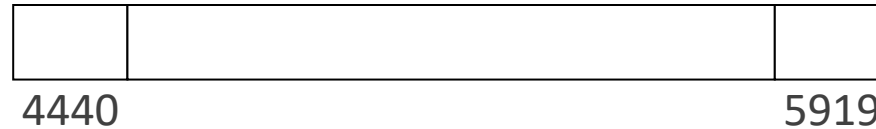
Fragment 2



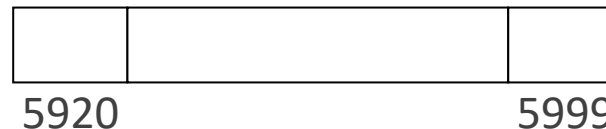
Fragment 3



Fragment 4



Fragment 5

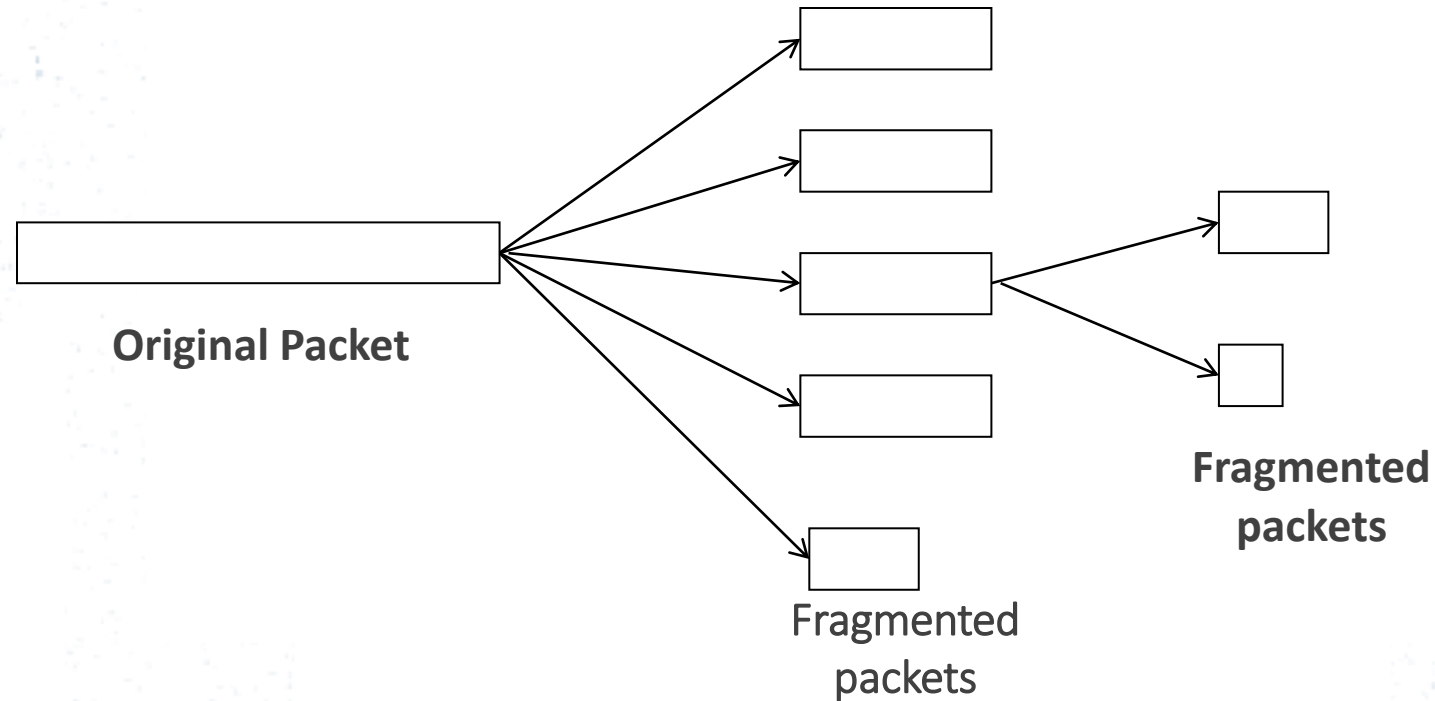


# Fragmentation offset cont.

- Offset value of fragment 1 =  $\frac{0}{8} = 0$
- Offset value of fragment 2 =  $\frac{1480}{8} = 185$
- Offset value of fragments 3 =  $\frac{2960}{8} = 370$
- Offset value of fragment 4 =  $\frac{4440}{8} = 555$
- Offset value of fragment 5 =  $\frac{5920}{8} = 740$

# Fragmentation offset cont.

- Fragmented packets travel independently
- They may travel through different routes to the destination
- While it is traveling it can be further fragmented at another intermediate network



# Fragmentation offset cont.

- Fragmented packets reach to the destination out of order.
- The fragmented packets are combined (defragmented) at the final destination by using the *OFFSET* values.



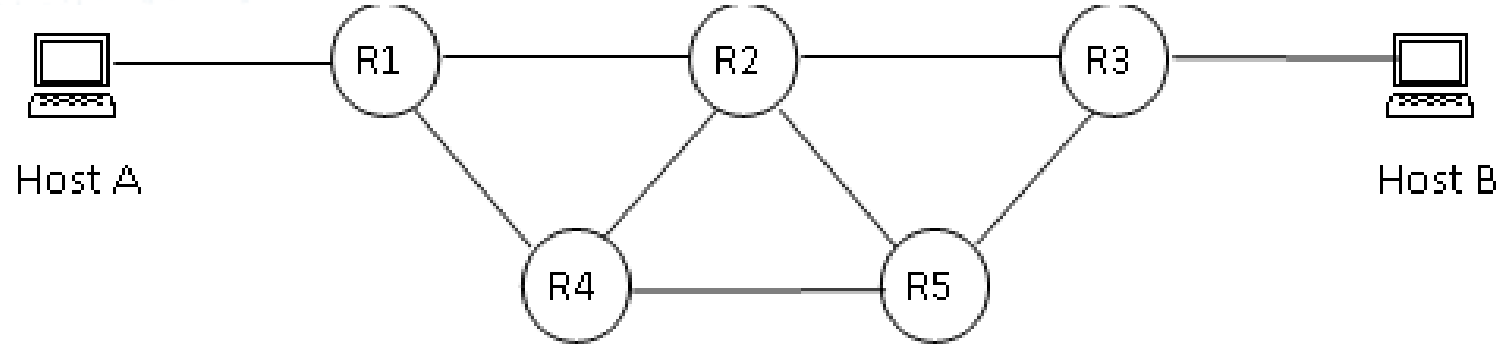
# Time To Live (TTL)

- IP packets travel through many routers in the network
- Each router routes the packet according to information in the routing table
- If there is a problem in a routing table the packet may be sent in a wrong direction and it can be randomly flow in the network. This kind of IP packets can even overload the network and finally crash the network
- In order to avoid such a situation, a parameter called “Time To Live” (TTL) is defined for each IP packet

# Time To Live (TTL) cont.

- TTL value can be initialized to any value at the transmitting router (A) (maximum is 255)
- The TTL value is decremented at each router by 1
- If the TTL value becomes zero at a router (B) , the packet will be discarded and an ICMP message is sent to the transmitted router(A) from the discarding router (B)

# Time To Live (TTL) cont.

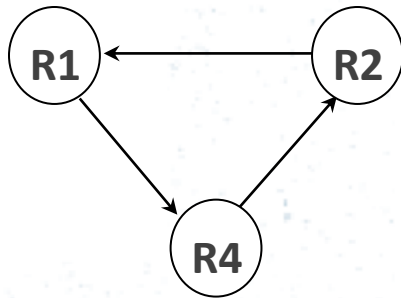


- Suppose host A sends an IP packet to Host B
- The TTL value is set to 6
- If the packet goes through Host A → R1 → R2 → R3 → Host B,

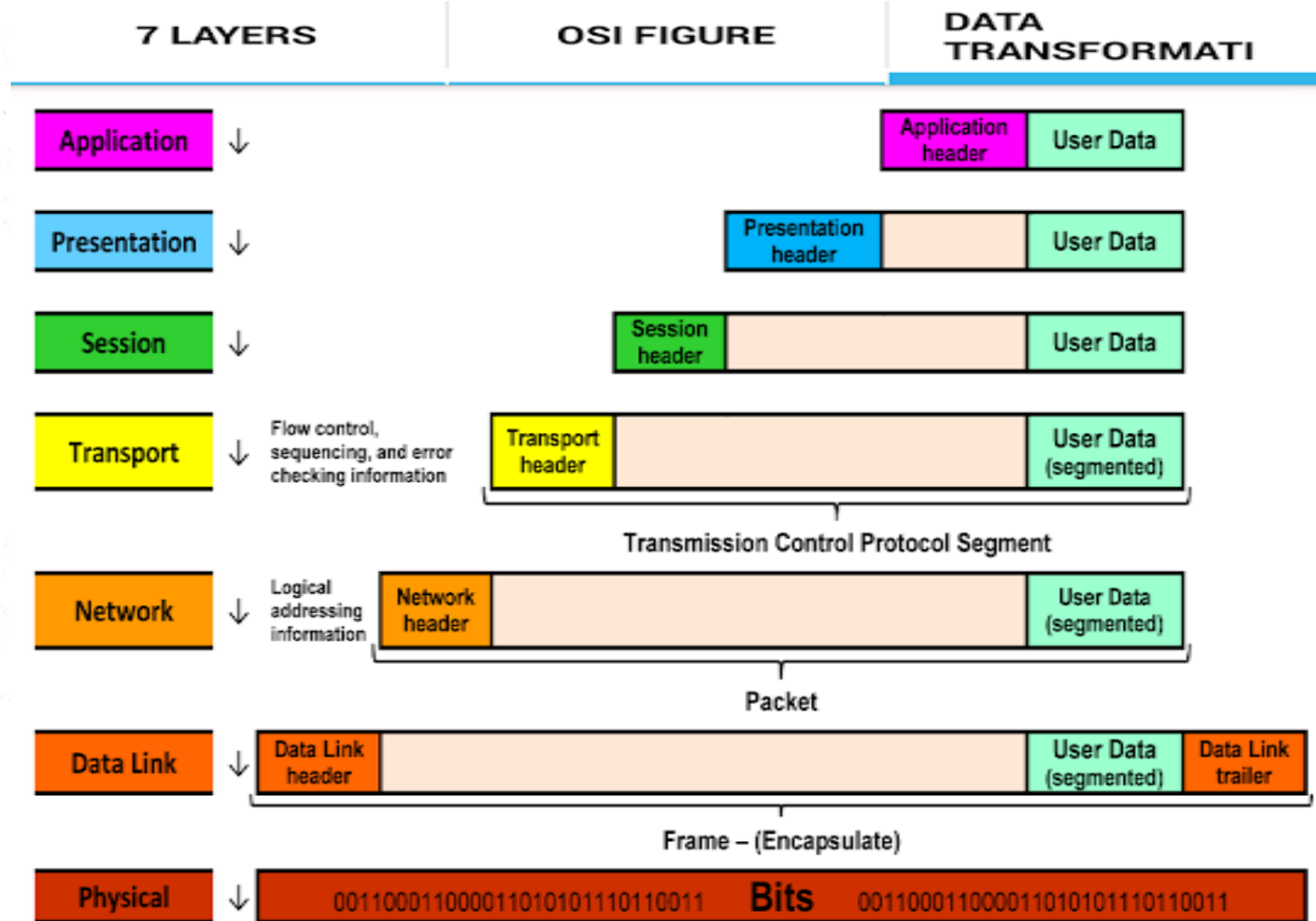
| Router | TTL Value |
|--------|-----------|
| R1     | = 5       |
| R2     | = 4       |
| R3     | = 3       |

# Time To Live (TTL) cont.

- Suppose there is a routing problem and the packet loops through the routes  $R1 \rightarrow R4 \rightarrow R2 \rightarrow R1 \rightarrow R4 \rightarrow R2 \rightarrow R1 \rightarrow R4 \rightarrow R2$

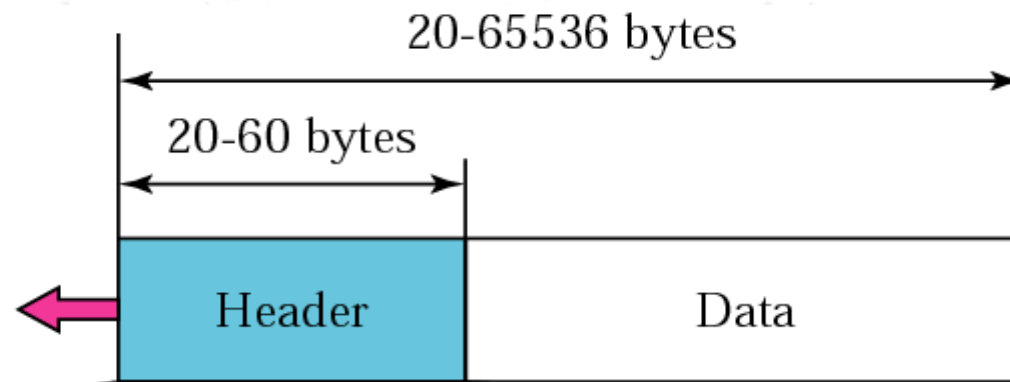


| Router | TTL Value   | Action   |
|--------|-------------|--|
| R1     | $6 - 1 = 5$ |  |
| R4     | $5 - 1 = 4$ |  |
| R2     | $4 - 1 = 3$ |  |
| R1     | $3 - 1 = 2$ |  |
| R4     | $2 - 1 = 1$ |  |
| R2     | $1 - 1 = 0$ | Discards the Packet<br>Send ICMP message to Host A |



Format of the information at each layer

# IP header



|                           |                |                    |                                 |  |
|---------------------------|----------------|--------------------|---------------------------------|--|
| VER<br>4 bits             | HLEN<br>4 bits | DS<br>8 bits       | Total length<br>16 bits         |  |
| Identification<br>16 bits |                | Flags<br>3 bits    | Fragmentation offset<br>13 bits |  |
| Time to live<br>8 bits    |                | Protocol<br>8 bits | Header checksum<br>16 bits      |  |
| Source IP address         |                |                    | 32 bits                         |  |
| Destination IP address    |                |                    | 32 bits                         |  |
| Option                    |                |                    | 32 bits                         |  |



# Version (VER)

- A field of 4 bits
- Indicates the version of the using IP addresses in the IP packet

IPv4 : 0100

IPv6 : 0110

# Header Length (HLEN)

- A 4 bit field indicates the number of 4 bytes in the header
- Header size in bytes = HLEN x 4

- The standard header size = 20 bytes

$$20 = 5 \times 4$$

$$\text{HLEN} = 5 \quad (0101)_2$$

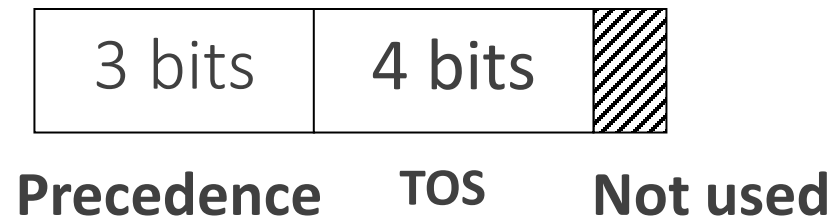
- The standard header size = 60 bytes

$$60 = 15 \times 4$$

$$\text{HLEN} = 15 \quad (1111)_2$$

# Service Type

- A 8 bit field
- IETF has changed the name of this field as **Differentiated Services**



- **Precedence** defines the priority of the packet
- **Precedence** is not used in IPv4

# Service Type cont.

- A 4-bit field
- Each bit has a special meaning
- There are five types of services

| <i>TOS Bits</i> | <i>Description</i>   |
|-----------------|----------------------|
| 0000            | Normal (default)     |
| 0001            | Minimize cost        |
| 0010            | Maximize reliability |
| 0100            | Maximize throughput  |
| 1000            | Minimize delay       |



# Service Type cont.

The application can select a specific type of service

| <i>Protocol</i> | <i>TOS Bits</i> | <i>Description</i>   |
|-----------------|-----------------|----------------------|
| ICMP            | 0000            | Normal               |
| BOOTP           | 0000            | Normal               |
| NNTP            | 0001            | Minimize cost        |
| IGP             | 0010            | Maximize reliability |
| SNMP            | 0010            | Maximize reliability |
| TELNET          | 1000            | Minimize delay       |
| FTP (data)      | 0100            | Maximize throughput  |
| FTP (control)   | 1000            | Minimize delay       |
| TFTP            | 1000            | Minimize delay       |
| SMTP (command)  | 1000            | Minimize delay       |
| SMTP (data)     | 0100            | Maximize throughput  |
| DNS (UDP query) | 1000            | Minimize delay       |
| DNS (TCP query) | 0000            | Normal               |
| DNS (zone)      | 0100            | Maximize throughput  |



# Total length

- A 16-bit field
- Gives the total length of the IP packet.

**Total length = data length + header length**

- If total length value is 300 and if this is a normal IP packet  
Header length = 20 bytes  
Data length =  $300 - 20$   
= 280 bytes

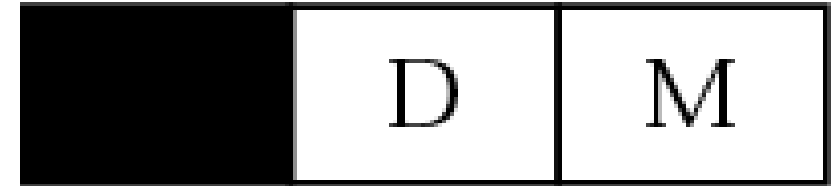


# Identification

- Each IP packet is identified by a serial number called “Identification”
- A 16 bit field
- The identification of each fragmented IP packet is equal to identification of original IP packet

# Flags

D: Do not fragment  
M: More fragments



- $D = 1$ , means is not allowed to be fragmented
- $D = 0$ , means is allowed to be fragmented
- $M = 0$ , means that there are no more fragments;  
The fragment is the last one.  
Non fragmented packet is considered the last fragment
- $M = 1$ , The fragment is not the last one.

# Fragmentation offset

- A 13-bit field
- This gives the offset value of the fragment

# Time To Live (TTL)

- A 8 bit field
- Defines the maximum number of hops the packet can travel

# Protocol

- The IP packet data can be UDP, TCP, ICMP, IGMP, EGP
- Used to identify the type of data a special field called “protocol”
- 8-bit field which defines the protocol number

| <i>Value</i> | <i>Protocol</i> |
|--------------|-----------------|
| 1            | ICMP            |
| 2            | IGMP            |
| 6            | TCP             |
| 17           | UDP             |
| 89           | OSPF            |

# Header Checksum

- A 16-bit field
- Checks the errors of the header only.
- If errors are found in the header, the whole IP packet is discarded.



# Source IP Address

- A 32-bit field
- This gives the IP address of the source

# Destination IP Address

- A 32-bit field
- This gives the IP address of the destination

