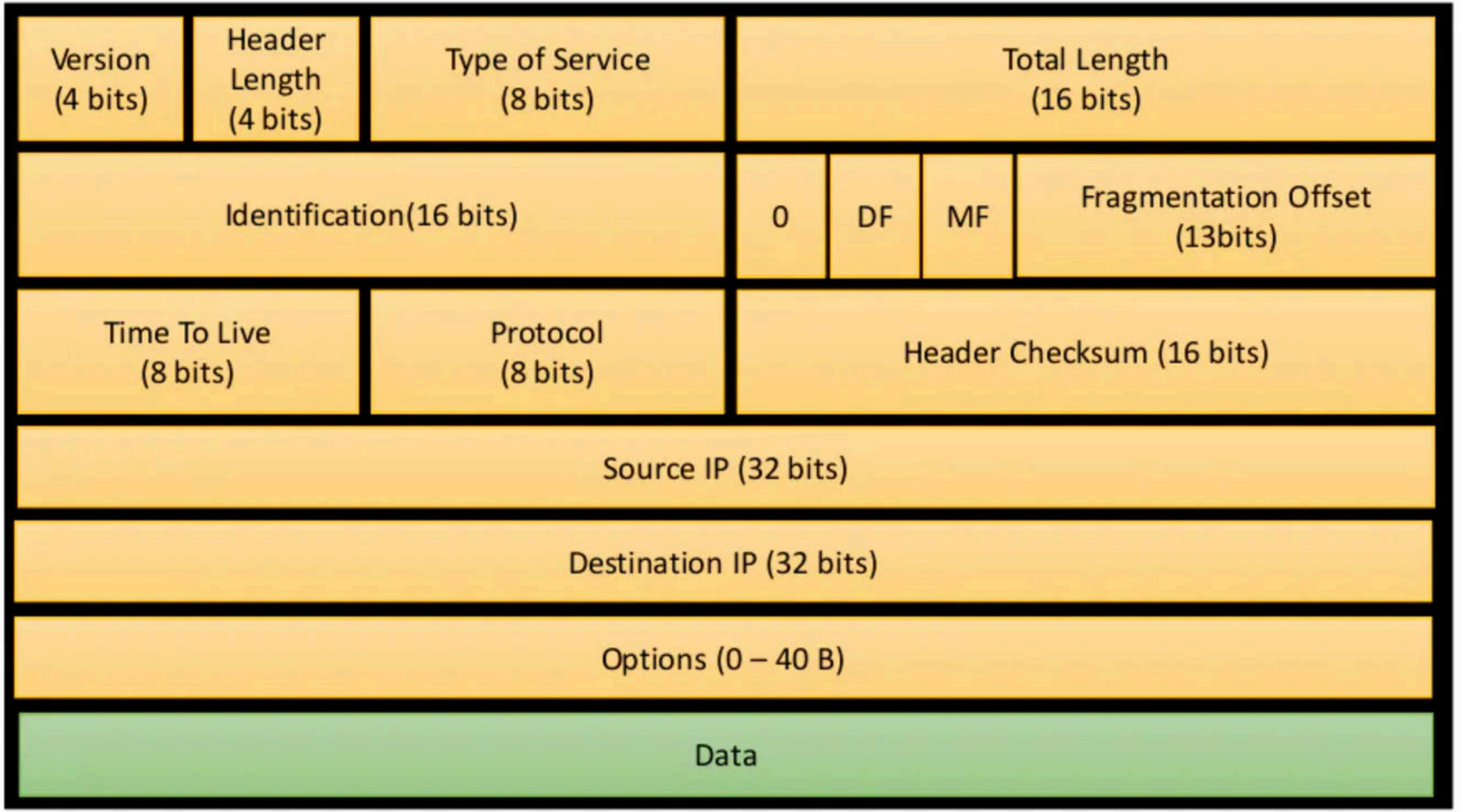
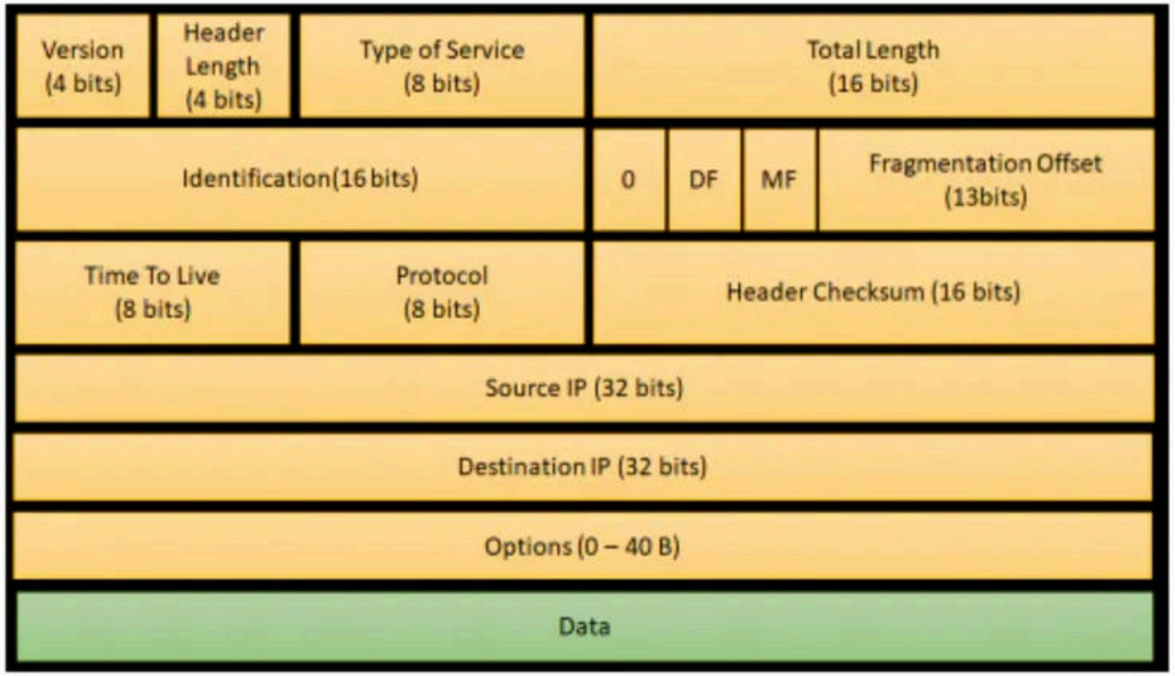
Computer Networks

IPV4


Version (4 bits)
Header Length (4 bits)
Type of Service (8 bits)
Total Length (16 bits)
Identification(16 bits)
0
DF
MF
Fragmentation Offset (13bits)
Time To Live (8 bits)
Protocol (8 bits)
Header Checksum (16 bits)
Source IP (32 bits)
Destination IP (32 bits)
Options (0 – 40 B)
Data

Version

•Version is a 4 bit field that indicates the IP version used.
•The most popularly used IP versions are version-4 (IPv4) and version-6 (IPv6).
•Only IPv4 uses the above header.
•So, this field always contains the decimal value 4.



Header Length

•Header length is a 4 bit field that contains the length of the IP header.
•It helps in knowing from where the actual data begins.

Header length = Header length field value x 4 bytes

Ex. If header length field contains decimal value 5 (represented as 0101), then-

Header length = 5 x 4 = 20 bytes

Minimum And Maximum Header Length-

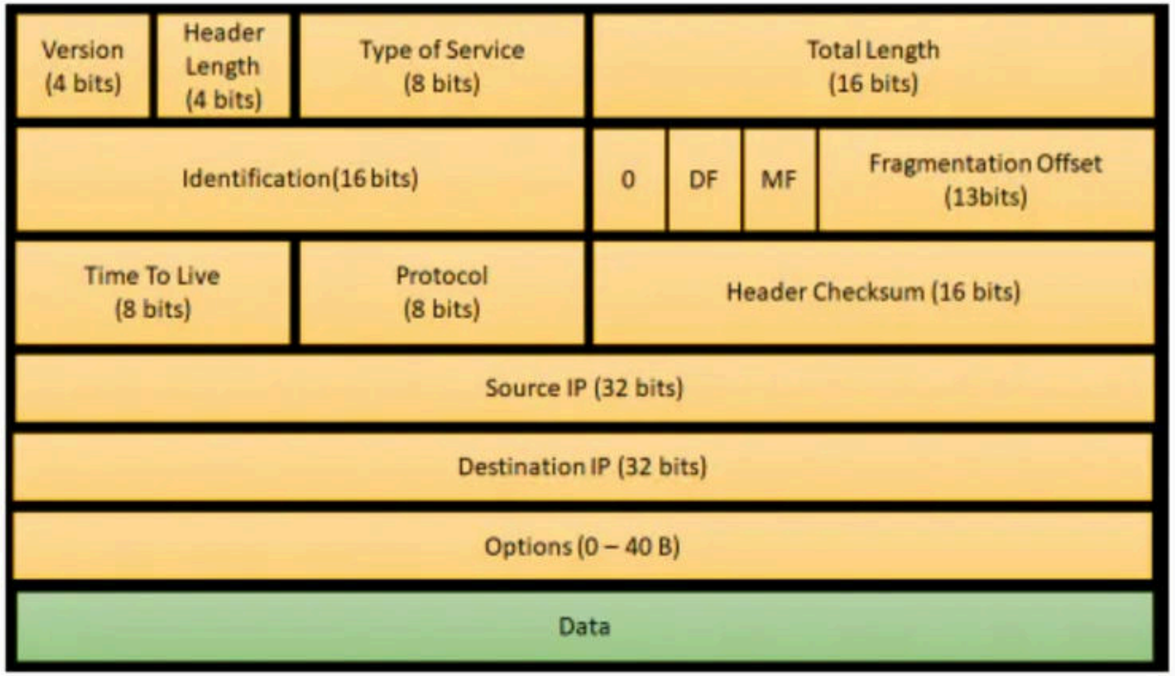
The length of IP header always lies in the range-

[20 bytes , 60 bytes]

•The initial 5 rows of the IP header are always used.
•So, minimum length of IP header = 5 x 4 bytes = 20 bytes.
•The size of the 6th row representing the Options field vary.
•The size of Options field can go up to 40 bytes.
•So, maximum length of IP header = 20 bytes + 40 bytes = 60 bytes.

Type Of Service

•Type of service is a 8 bit field that is used for Quality of Service (QoS).
•The datagram is marked for giving a certain treatment using this field.



Total Length

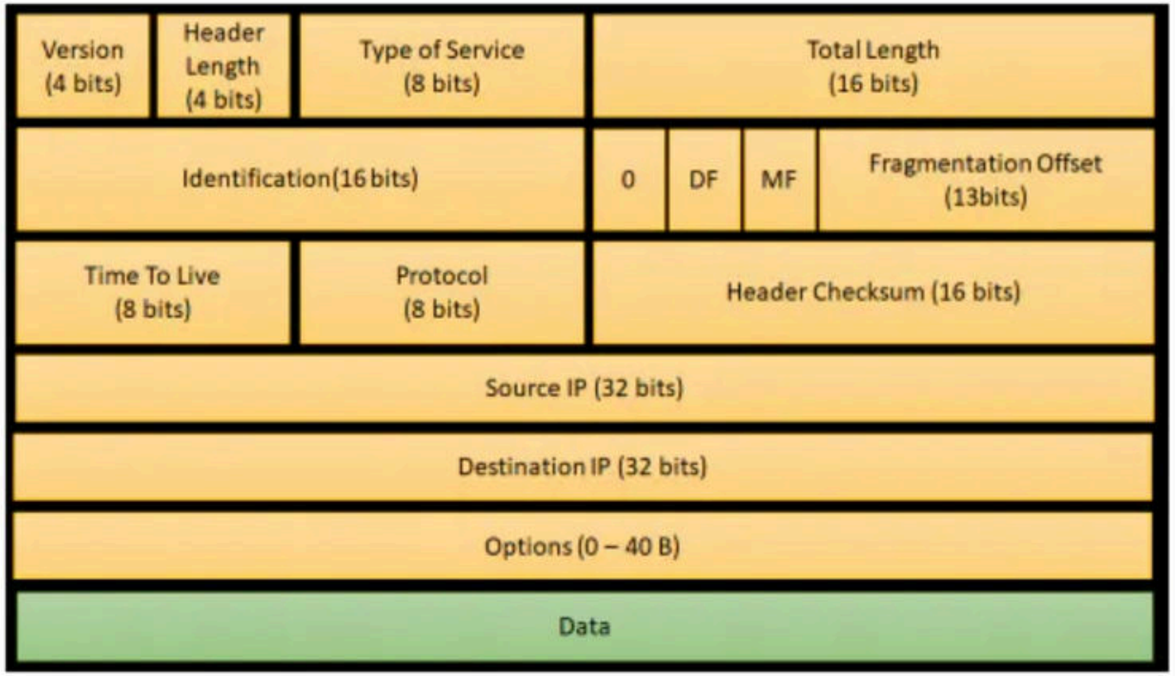
•Total length is a 16 bit field that contains the total length of the datagram (in bytes).

Total length = Header length + Payload length

•Minimum total length of datagram = 20 bytes (20 bytes header + 0 bytes data)
•Maximum total length of datagram = Maximum value of 16 bit word = 65535 bytes

Identification

•Identification is a 16 bit field.
•It is used for the identification of the fragments of an original IP datagram.



DF Bit

•DF bit stands for Do Not Fragment bit.
•Its value may be 0 or 1.

When DF bit is set to 0,

•It grants the permission to the intermediate devices to fragment the datagram if required.

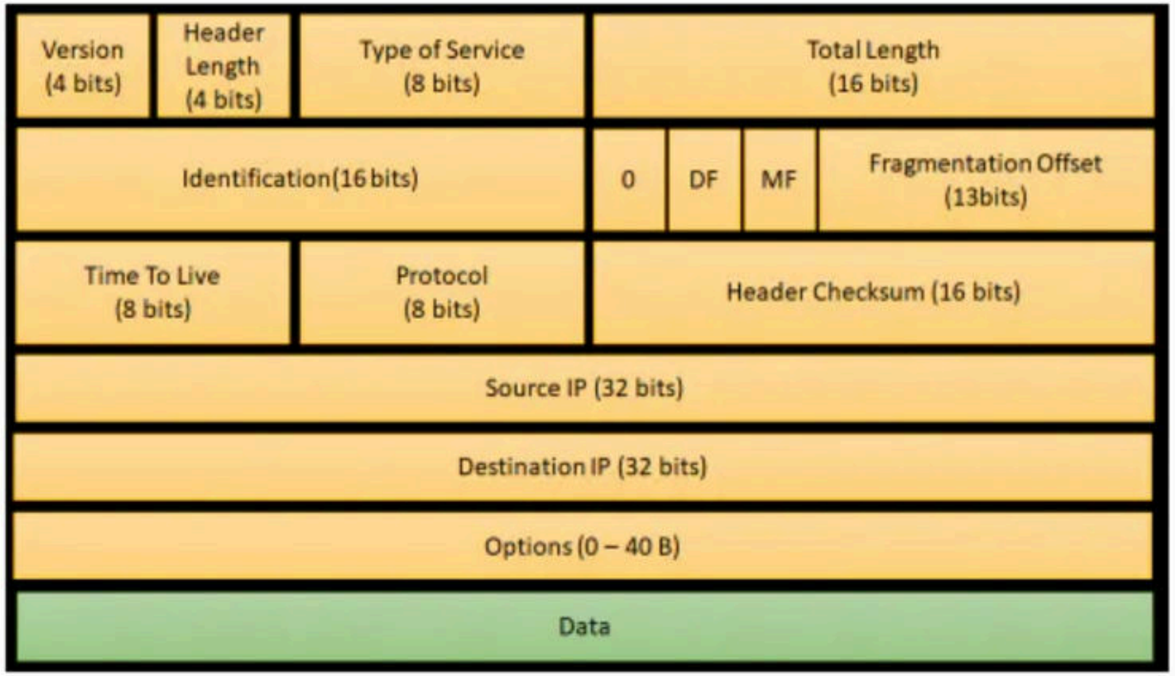
When DF bit is set to 1,
•It indicates the intermediate devices not to fragment the IP datagram at any cost.
•If network requires the datagram to be fragmented to travel further but settings does not allow its fragmentation, then it is discarded.
•An error message is sent to the sender saying that the datagram has been discarded due to its settings.

MF Bit

•MF bit stands for More Fragments bit.
•Its value may be 0 or 1.

When MF bit is set to 0,
•It indicates to the receiver that the current datagram is either the last fragment in the set or that it is the only fragment.

When MF bit is set to 1,
•It indicates to the receiver that the current datagram is a fragment of some larger datagram.
•More fragments are following.
•MF bit is set to 1 on all the fragments except the last one.



Fragment Offset

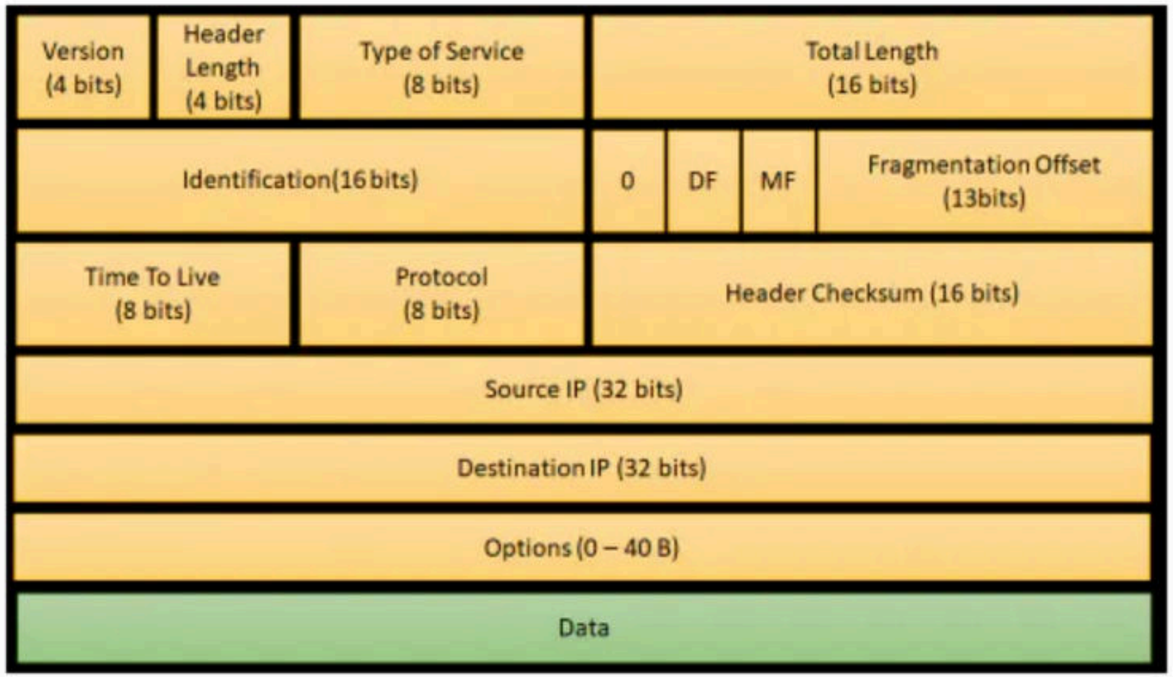
•Fragment Offset is a 13 bit field.
•It indicates the position of a fragmented datagram in the original unfragmented IP datagram.
•The first fragmented datagram has a fragment offset of zero.

Fragment offset for a given fragmented datagram
= Number of data bytes ahead of it in the original unfragmented datagram

Time To Live

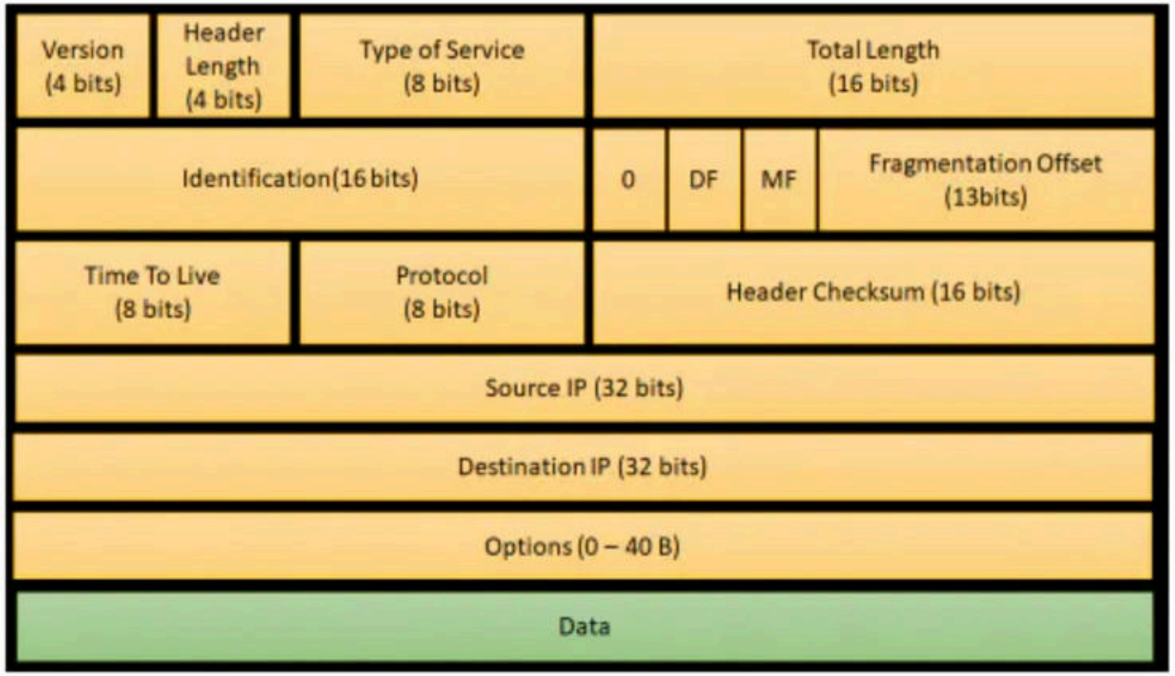
•Time to live (TTL) is a 8 bit field.
•It indicates the maximum number of hops a datagram can take to reach the destination.
•The main purpose of TTL is to prevent the IP datagrams from looping around forever in a routing loop.

The value of TTL is decremented by 1 when-
•Datagram takes a hop to any intermediate device having network layer.
•Datagram takes a hop to the destination.



Protocol

•Protocol is a 8 bit field.
•It tells the network layer at the destination host to which protocol the IP datagram belongs to.
•In other words, it tells the next level protocol to the network layer at the destination side.
•Protocol number of ICMP is 1, IGMP is 2, TCP is 6 and UDP is 17.



Why Protocol Number Is A Part Of IP Header?

Consider-
•An IP datagram is sent by the sender to the receiver.
•When datagram reaches at the router, it's buffer is already full.

In such a case,
•Router does not discard the datagram directly.
•Before discarding, router checks the next level protocol number mentioned in its IP header.
•If the datagram belongs to TCP, then it tries to make room for the datagram in its buffer.
•It creates a room by eliminating one of the datagrams having lower priority.
•This is because it knows that TCP is a reliable protocol and if it discards the datagram, then it will be sent again by the sender.
•The order in which router eliminate the datagrams from its buffer is-

ICMP > IGMP > UDP > TCP

If protocol number would have been inside the datagram, then-
•Router could not look into it.
•This is because router has only three layers- physical layer, data link layer and network layer.

That is why, protocol number is made a part of IP header.

Header Checksum

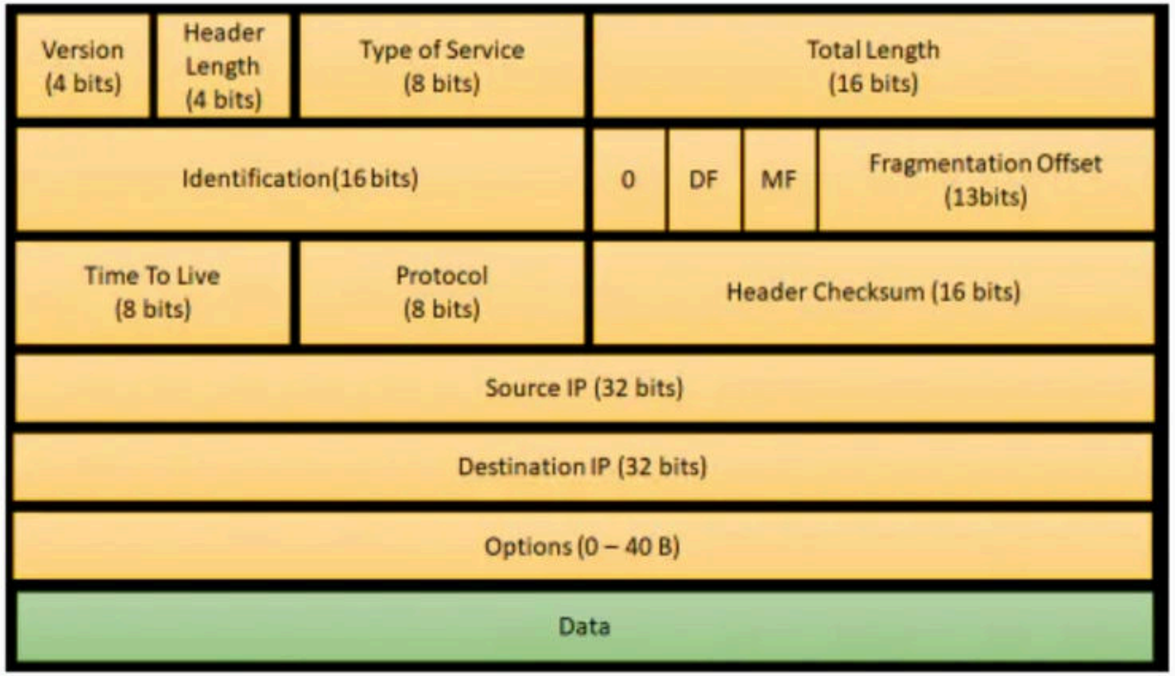
- Header checksum is a 16 bit field.
- It contains the checksum value of the entire header.
- The checksum value is used for error checking of the header.

At each hop,

- The header checksum is compared with the value contained in this field.
- If header checksum is found to be mismatched, then the datagram is discarded.
- Router updates the checksum field whenever it modifies the datagram header.

The fields that may be modified are-

1. TTL
2. Options
3. Datagram Length
4. Header Length
5. Fragment Offset



Source IP Address

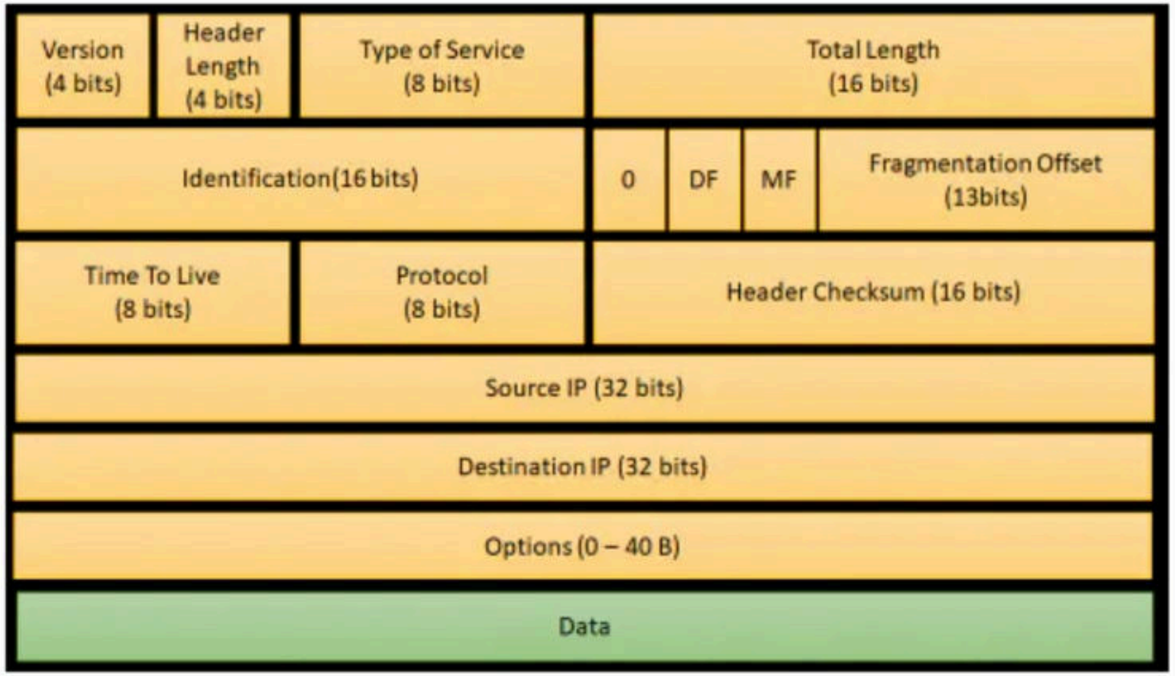
•Source IP Address is a 32 bit field.
•It contains the logical address of the sender of the datagram.

Destination IP Address

•Destination IP Address is a 32 bit field.
•It contains the logical address of the receiver of the datagram.

Options

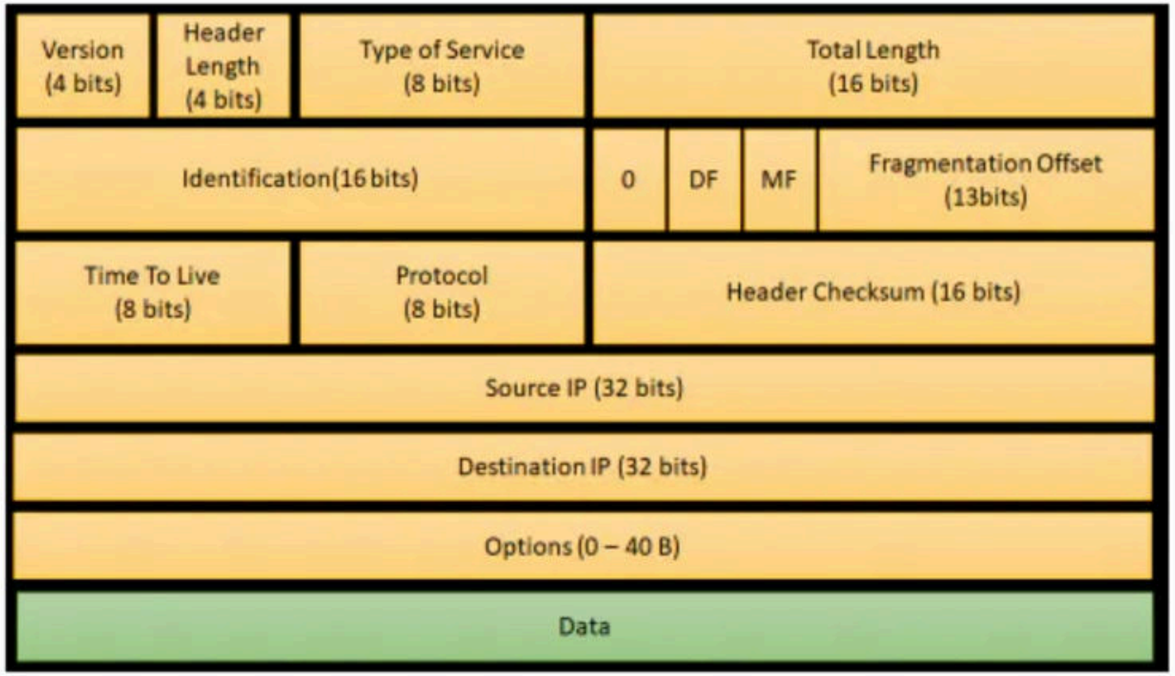
•Options is a field whose size vary from 0 bytes to 40 bytes.
•This field is used for several purposes such as-
1.Record route
2.Source routing
3.Padding



1. Record Route-

•A record route option is used to record the IP Address of the routers through which the datagram passes on its way.
•When record route option is set in the options field, IP Address of the router gets recorded in the Options field.

The maximum number of IPv4 router addresses that can be recorded in the Record Route option field of an IPv4 header is 9.

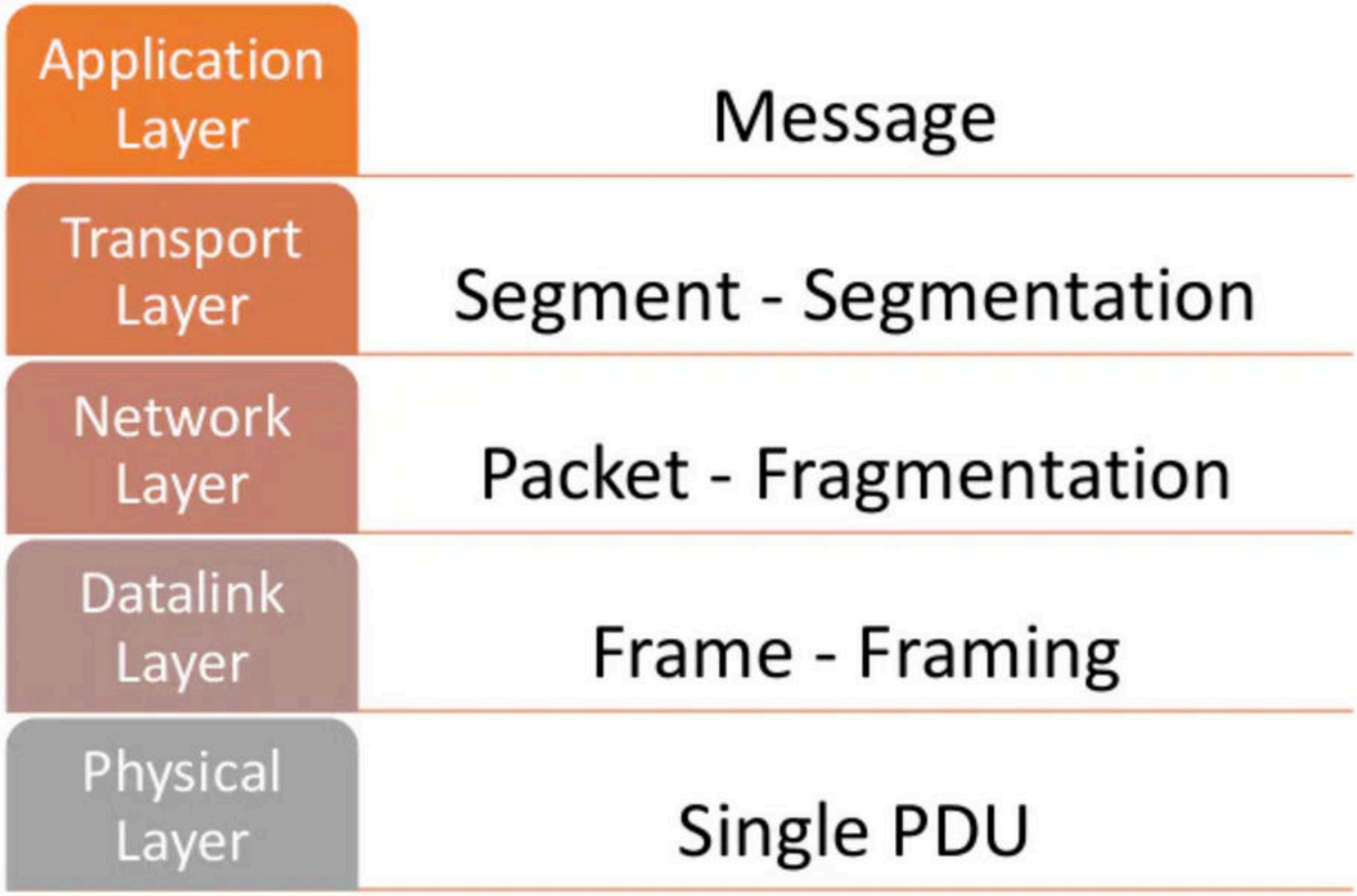


2. Source Routing-

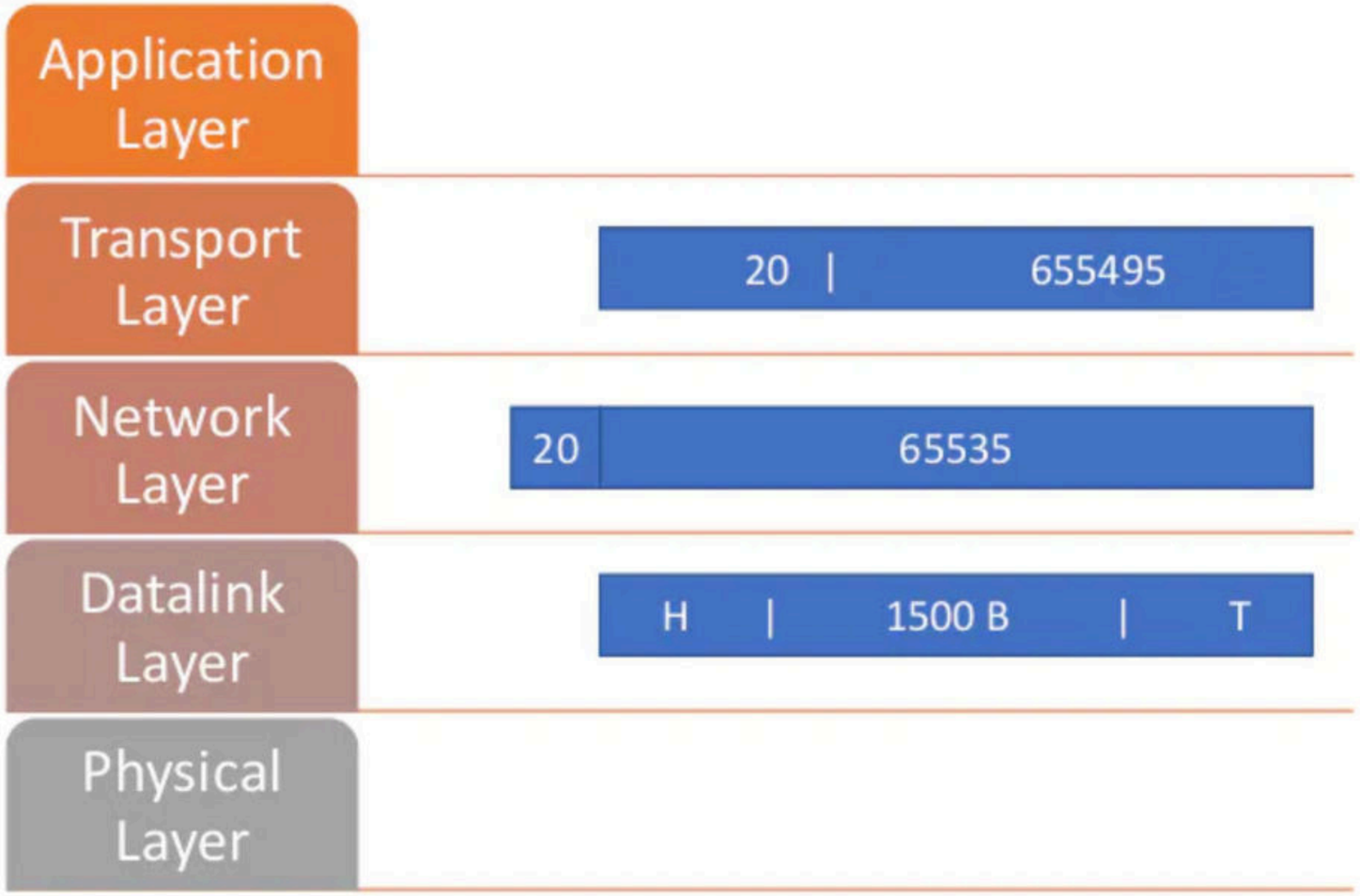
•A source routing option is used to specify the route that the datagram must take to reach the destination.
•This option is generally used to check whether a certain path is working fine or not.
•Source routing may be loose or strict.

3. Padding-

•Addition of dummy data to fill up unused space in the transmission unit and make it conform to the standard size is called as padding.
•Options field is used for padding.


Application Layer
Message
Transport Layer
Segment - Segmentation
Network Layer
Packet - Fragmentation
Datalink Layer
Frame - Framing
Physical Layer
Single PDU

Example: Segmentation and Fragmentation



SEGMENTATION AND FRAGMENTATION

This occurs during the original creation of the packets when a set of data doesn't fit within the "Maximum Segment Size (MSS)".
The data is then divided into multiple segments referred as "Protocol Data Unit". This process is known as **Segmentation.**

In order to avoid Fragmentation (which we will see further) , note that
(Number of bytes in the data segment + the header) < MTU

Fragmentation occurs during the original creation of frames where the network layer must send packets down to the Data Link Layer for transmission. Some Data Link Layer technologies have limits on the length of the data that can be sent. In short some links have smaller MTU (Maximum Transmission Unit).
If the packet that is to be sent is larger than the MTU then it is divided into pieces.
This process is known as fragmentation.
These pieces are reassembled once they arrive at the network layer of the destination.
As mentioned earlier Fragmentation can be avoided if,
(Number of bytes in the data segment + the header) < MTU

Computer Networks

Reassembly Algorithm

Reassembly Algorithm

Receiver applies the following steps for reassembly of all the fragments-

1.It identifies whether datagram is fragmented or not using MF bit and Fragment offset field.

2.It identifies all the fragments belonging to the same datagram using identification field.

3.It identifies the first fragment. Fragment with offset field value = 0 is the first fragment.

4.It identifies the subsequent fragments using total length, header length and fragment offset.

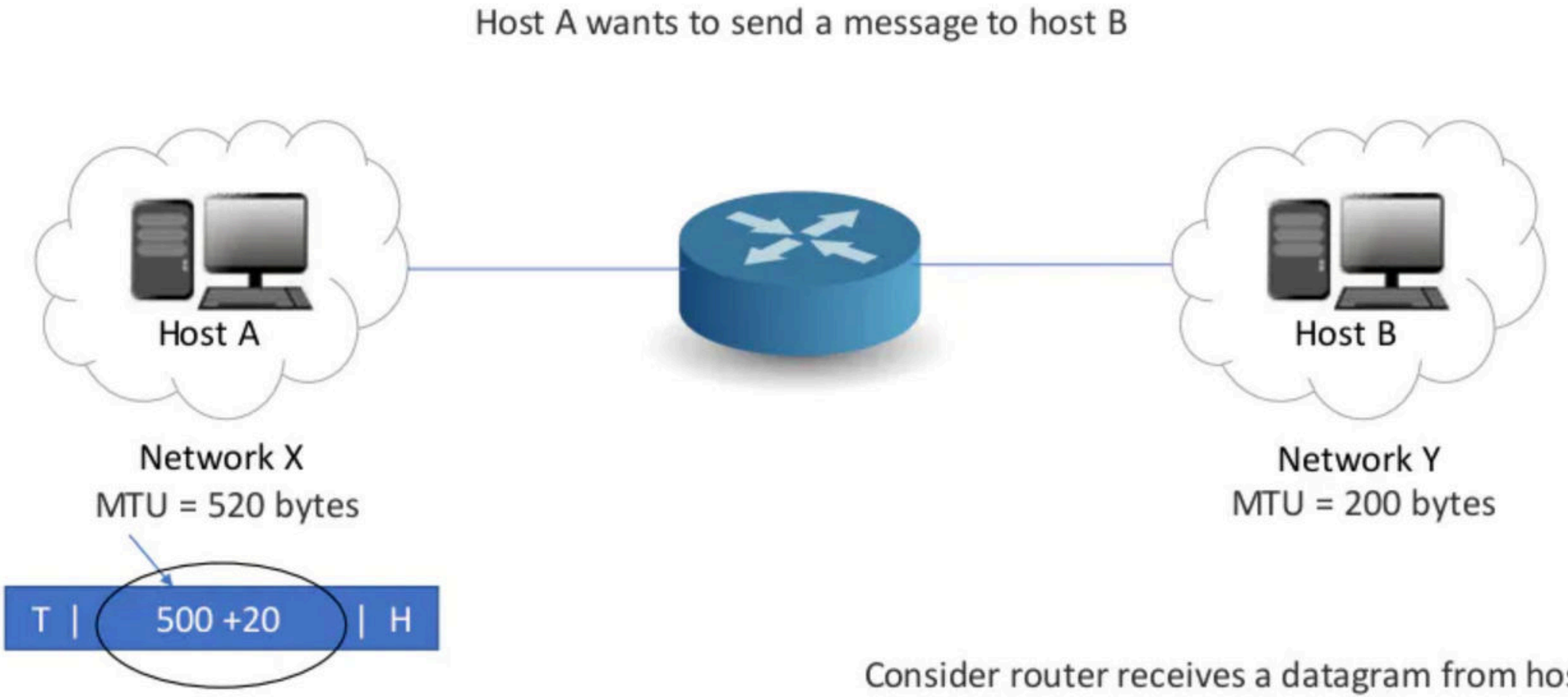
5.It repeats step-04 until MF bit = 0.

Computer Networks

Fragmentation

Lets us discuss some examples of IP fragmentation to understand how the fragmentation is actually carried out.

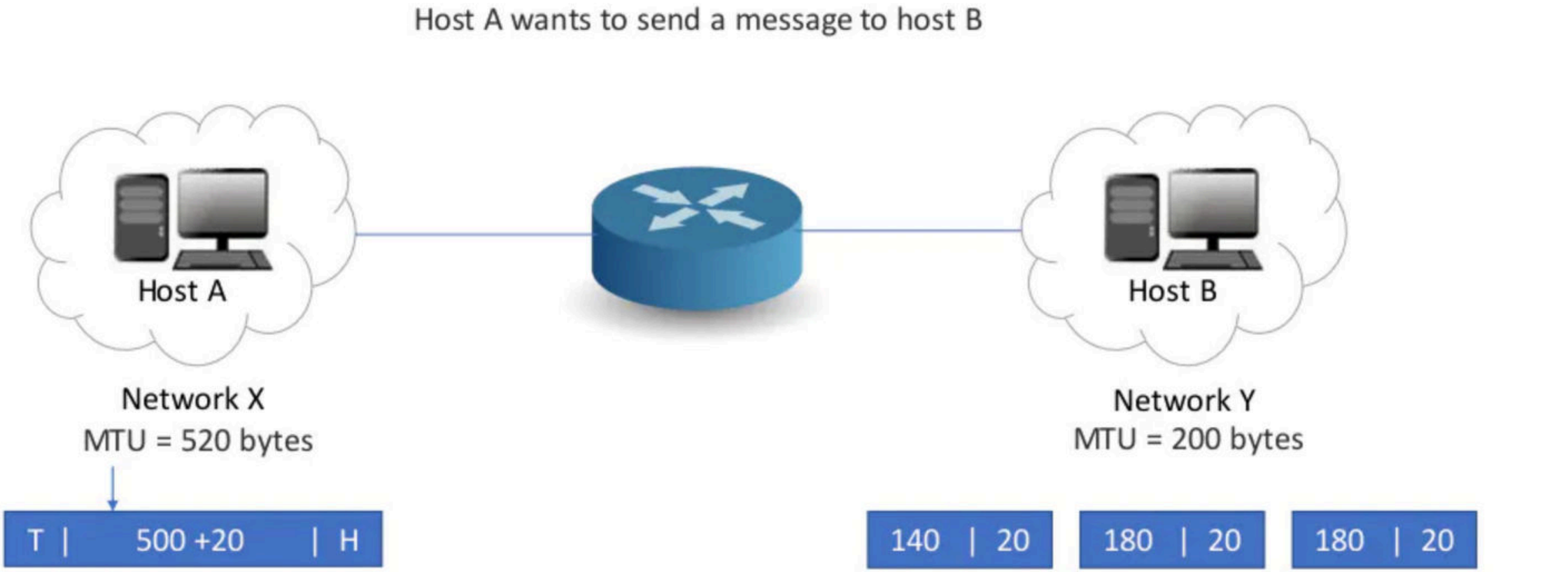
EXAMPLE 1



Consider router receives a datagram from host A having-
Header length = 20 bytes
Payload length = 500 bytes
Total length = 520 bytes

Lets us discuss some examples of IP fragmentation to understand how the fragmentation is actually carried out.

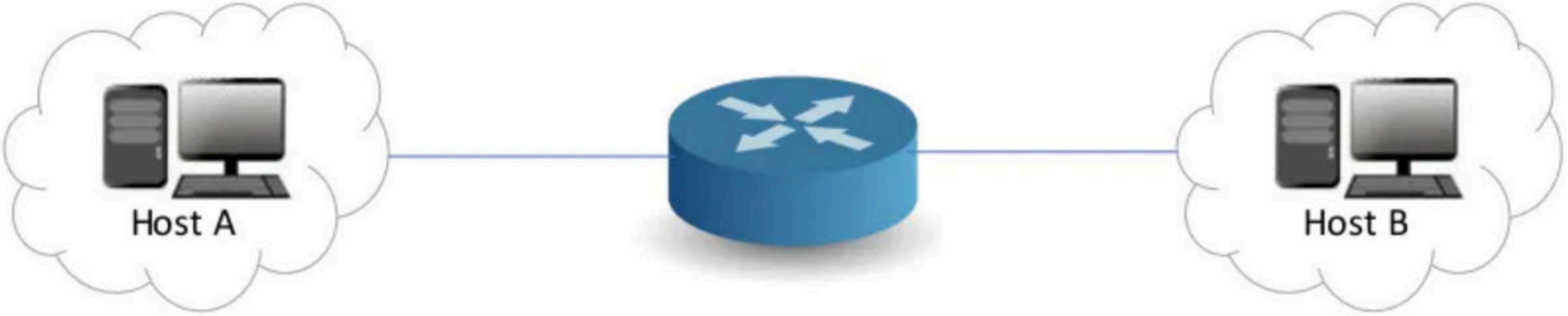
EXAMPLE 1



The amount of data sent in one fragment is chosen such that-
It is as large as possible but less than or equal to MTU.
It is a multiple of 8 so that pure decimal value can be obtained for the fragment offset field.

NOTE

- It is not compulsory for the last fragment to contain the amount of data that is a multiple of 8.
- This is because it does not have to decide the fragment offset value for any other fragment.



Network X
MTU = 520 bytes

500 +20

Network Y
MTU = 200 bytes

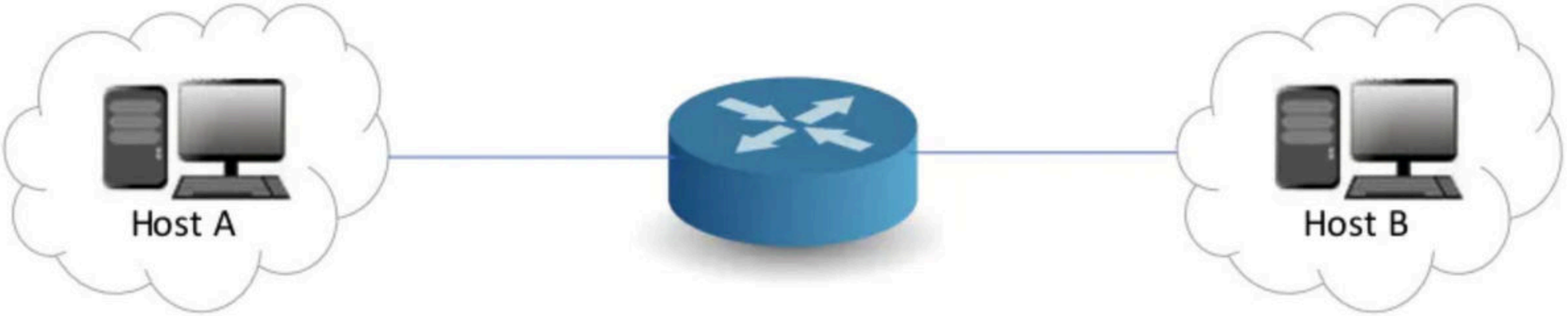
Following the above rule,
Router decides to send maximum 176 bytes of data in one fragment.
This is because it is the greatest value that is a multiple of 8 and less than MTU.

Router creates three fragments of the original datagram where-

First fragment contains the data = 176 bytes

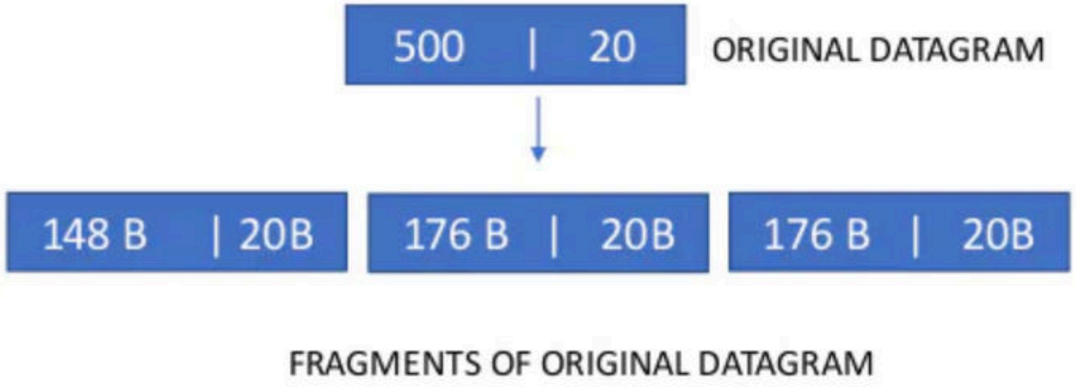
Second fragment contains the data = 176 byes

Third fragment contains the data = 148 bytes



Network X
MTU = 520 bytes

Network Y
MTU = 200 bytes



The information contained in the IP header of each fragment is-

Header Information Of 1st Fragment-

- Header length field value = 20 / 4 = 5
- Total length field value = 176 + 20 = 196
- MF bit = 1
- Fragment offset field value = 0
- Header checksum is recalculated.
- Identification number is same as that of original datagram.

Header Information Of 2nd Fragment-

- Header length field value = 20 / 4 = 5
- Total length field value = 176 + 20 = 196
- MF bit = 1
- Fragment offset field value = 176 / 8 = 22
- Header checksum is recalculated.
- Identification number is same as that of original datagram.

Header Information Of 3rd Fragment-

- Header length field value = 20 / 4 = 5
- Total length field value = 148 + 20 = 168
- MF bit = 0
- Fragment offset field value = (176 + 176) / 8 = 44
- Header checksum is recalculated.
- Identification number is same as that of original datagram.

Router transmits all the fragments.

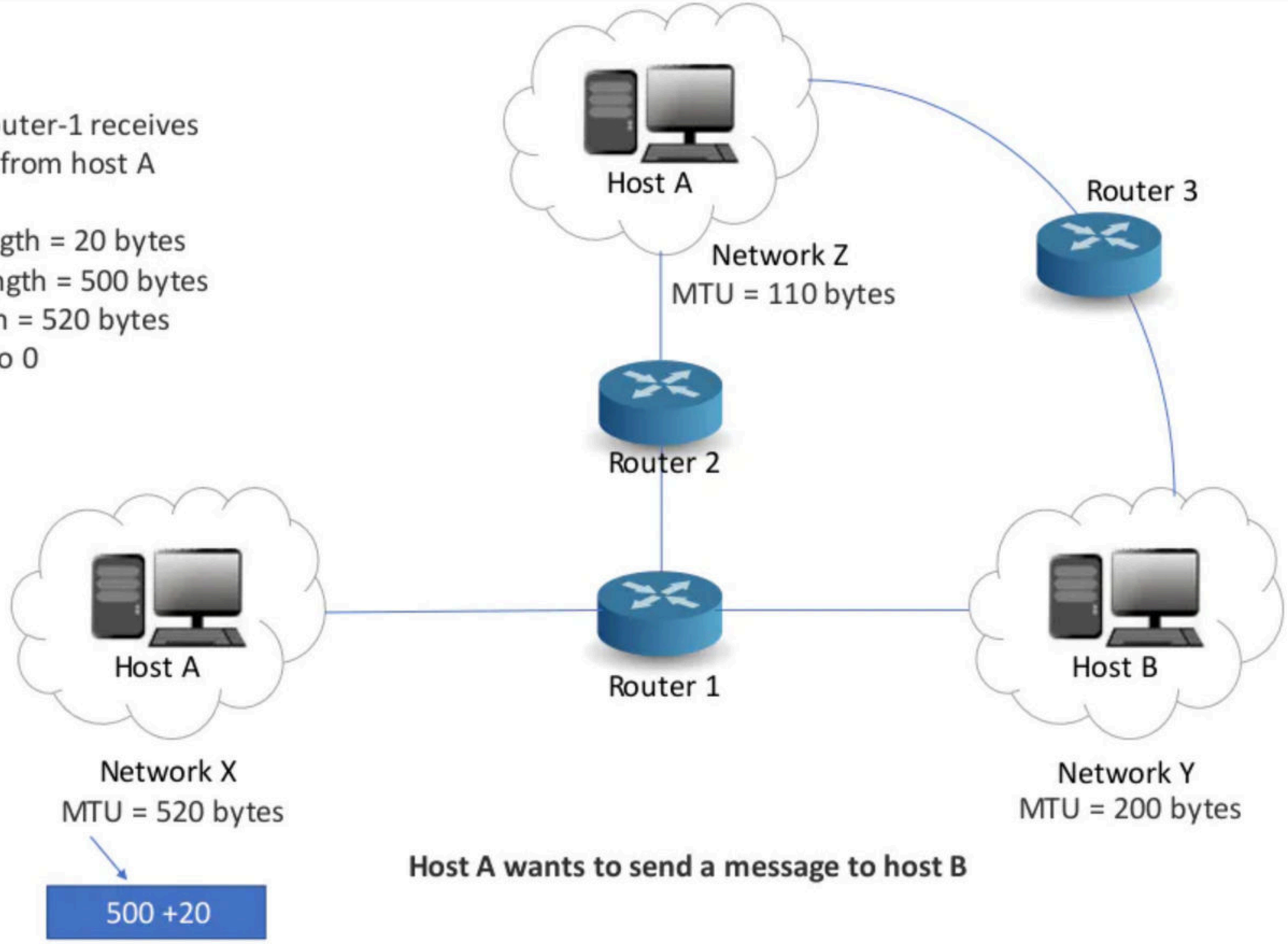
At destination side,

- Receiver receives 3 fragments of the datagram.
- Reassembly algorithm is applied to combine all the fragments to obtain the original datagram.

EXAMPLE 2

Consider Router-1 receives a datagram from host A having-

- Header length = 20 bytes
- Payload length = 500 bytes
- Total length = 520 bytes
- DF bit set to 0

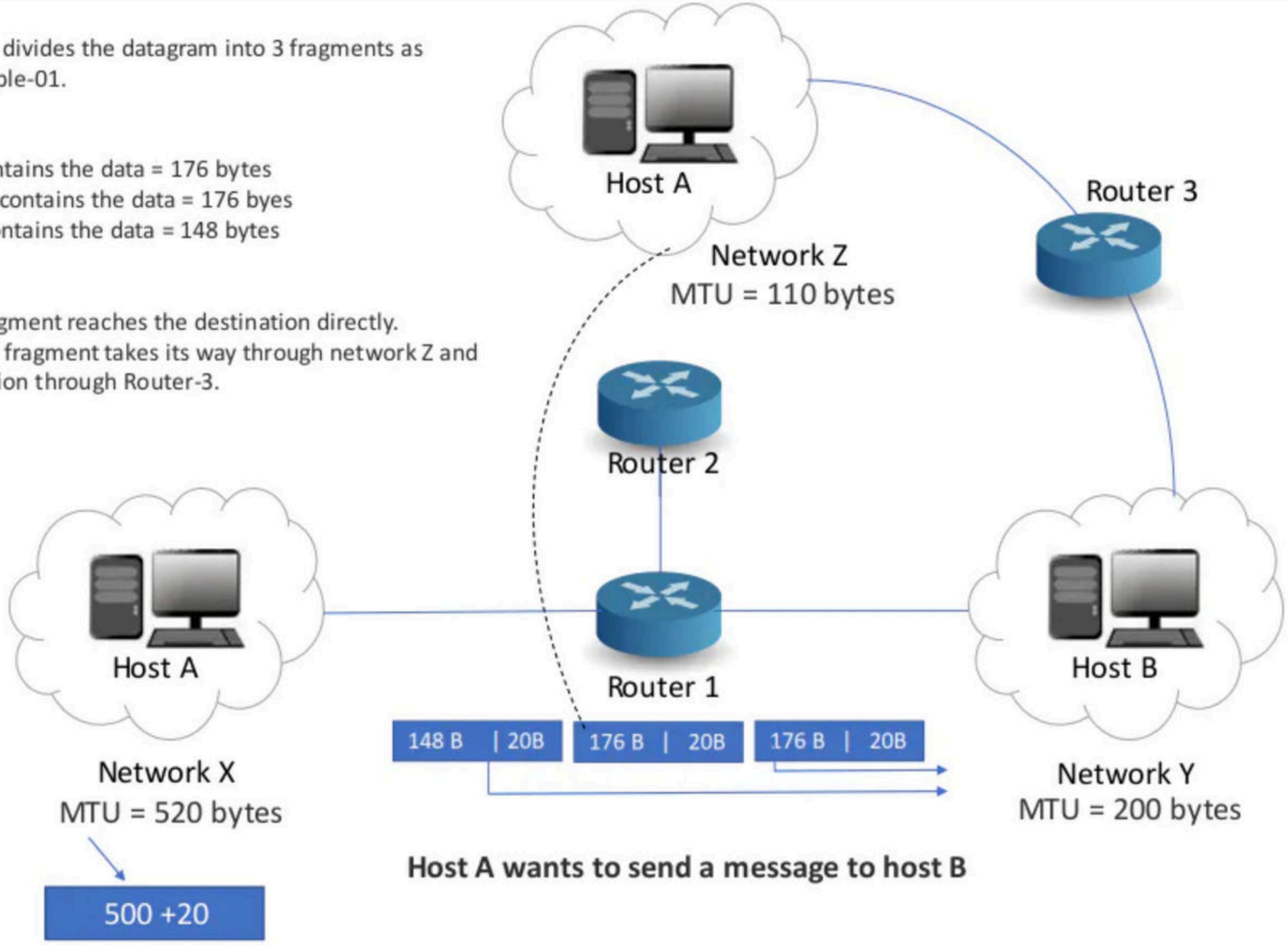


Host A wants to send a message to host B

Consider Router-1 divides the datagram into 3 fragments as discussed in Example-01.

Then,
- First fragment contains the data = 176 bytes
- Second fragment contains the data = 176 byes
- Third fragment contains the data = 148 bytes

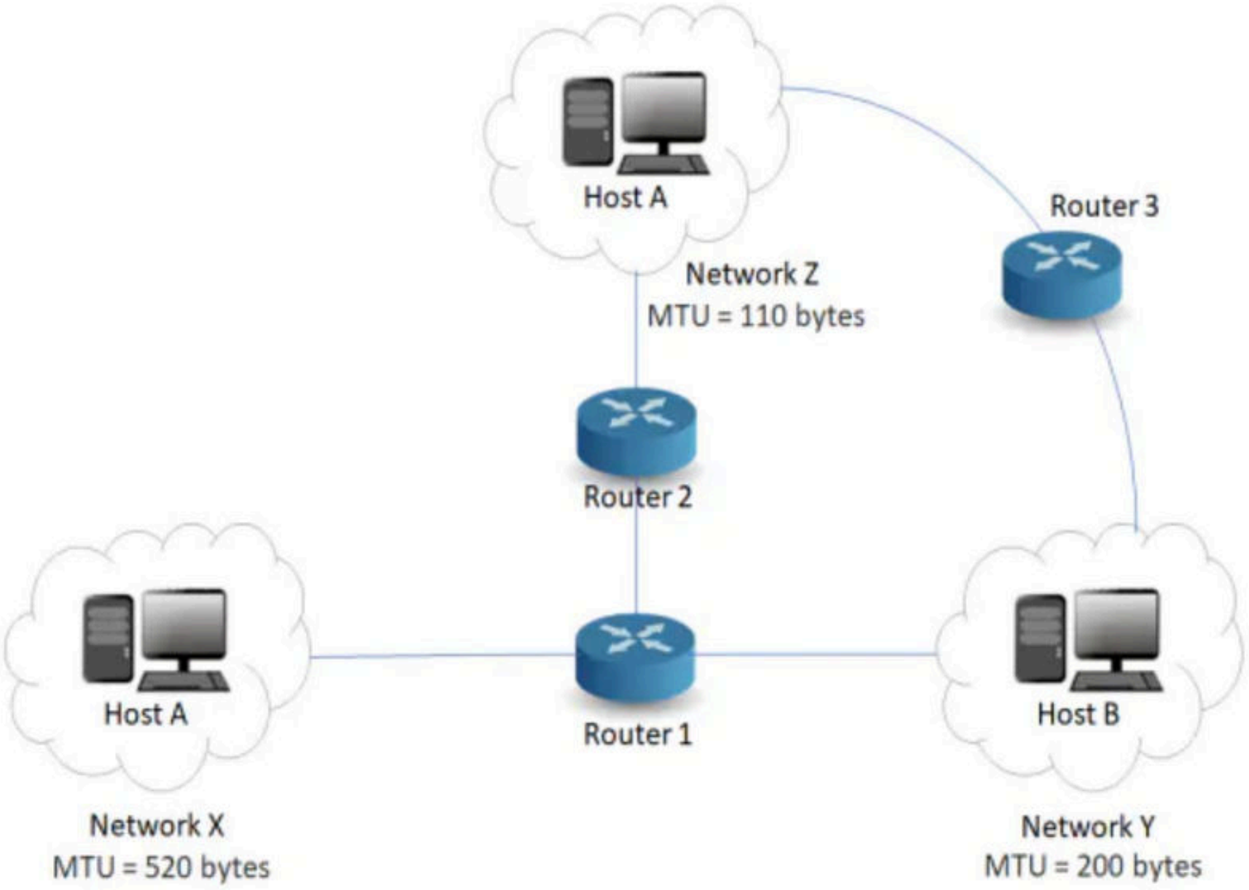
Now, consider-
- First and third fragment reaches the destination directly.
- However, second fragment takes its way through network Z and reach the destination through Router-3.



Now, let us discuss the journey of fragment-2 and how it finally reaches the destination.

Router-2 receives a datagram (second fragment of original datagram) where-

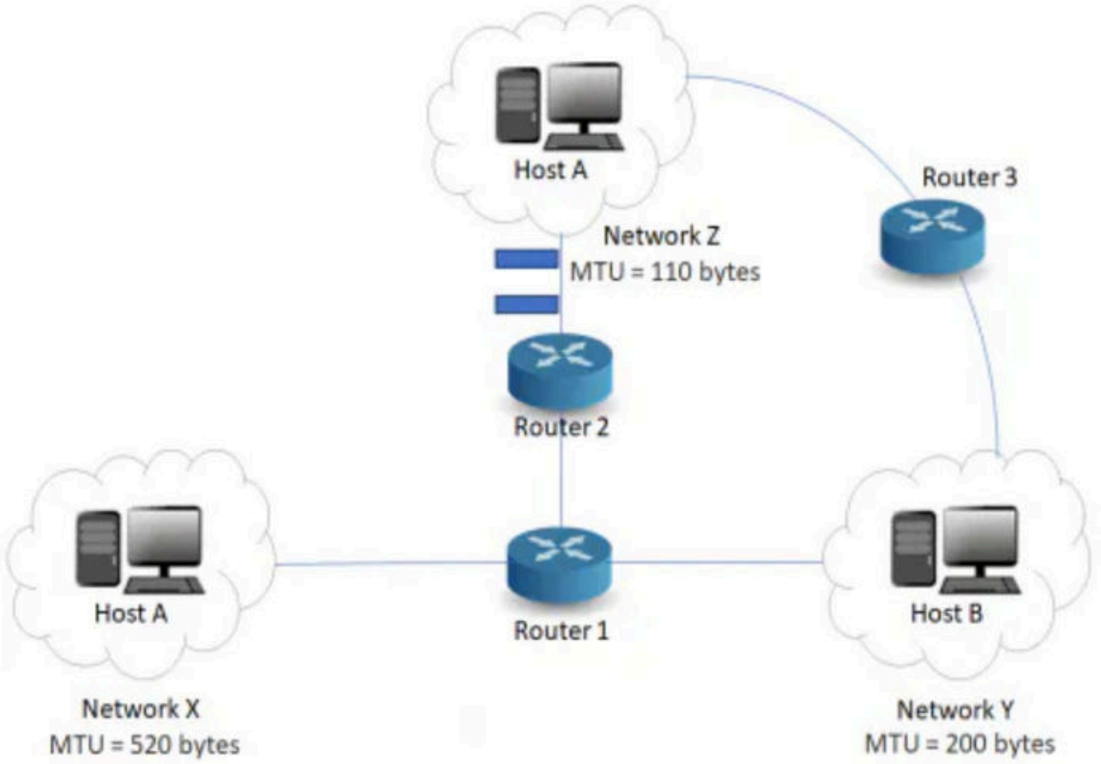
- Header length = 20 bytes
- Payload length = 176 bytes
- Total length = 196 bytes
- DF bit set to 0



Step-01:

Router-2 examines the datagram and finds-
•Size of the datagram = 196 bytes
•Destination is network Z having MTU = 110 bytes
•DF bit is set to 0

Router-2 concludes-
•Size of the datagram is greater than MTU.
•So, it will have to divide the datagram into fragments.
•DF bit is set to 0.
•So, it is allowed to create fragments of the datagram.



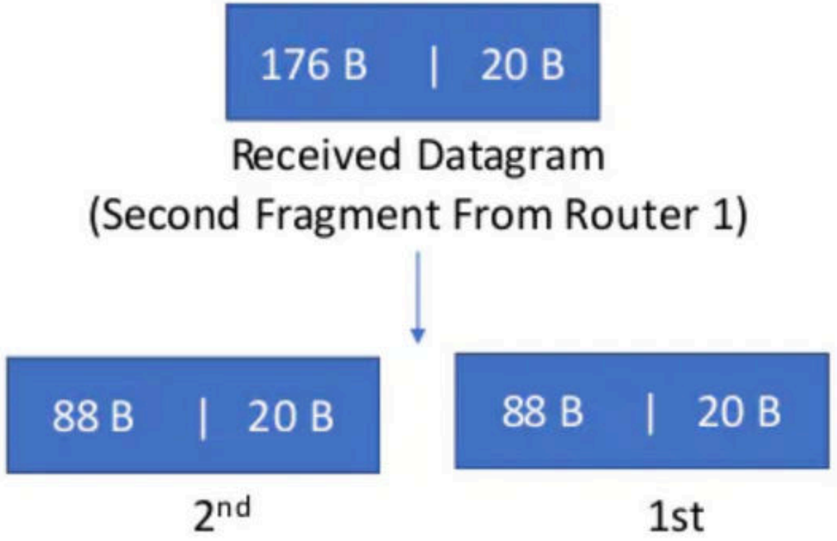
Step-02:

Router-2 decides the amount of data that it should transmit in each fragment.

Router-2 knows-
•MTU of the destination network = 110 bytes.
•So, maximum total length of any fragment can be only 110 bytes.
•Out of 110 bytes, 20 bytes will be taken by the header.
•So, maximum amount of data that can be sent in any fragment = 90 bytes.

Following the rule,
•Router-2 decides to send maximum 88 bytes of data in one fragment.
•This is because it is the greatest value that is a multiple of 8 and less than MTU.

Router-2 creates two fragments of the received datagram where-



The information contained in the IP header of each fragment is-

Header Information Of 1st Fragment-

- Header length field value = 20 / 4 = 5
- Total length field value = 88 + 20 = 108
- MF bit = 1
- Fragment offset field value = 176 / 8 = 22
- Header checksum is recalculated.
- Identification number is same as that of original datagram.

NOTE-

- This fragment is **NOT** the first fragment of the original datagram.
- It is the first fragment of the datagram received by Router-2.
- The datagram received by Router-2 is the second fragment of the original datagram.
- This datagram will serve as the second fragment of the original datagram.
- Therefore, fragment offset field is set according to the first fragment of the original datagram.

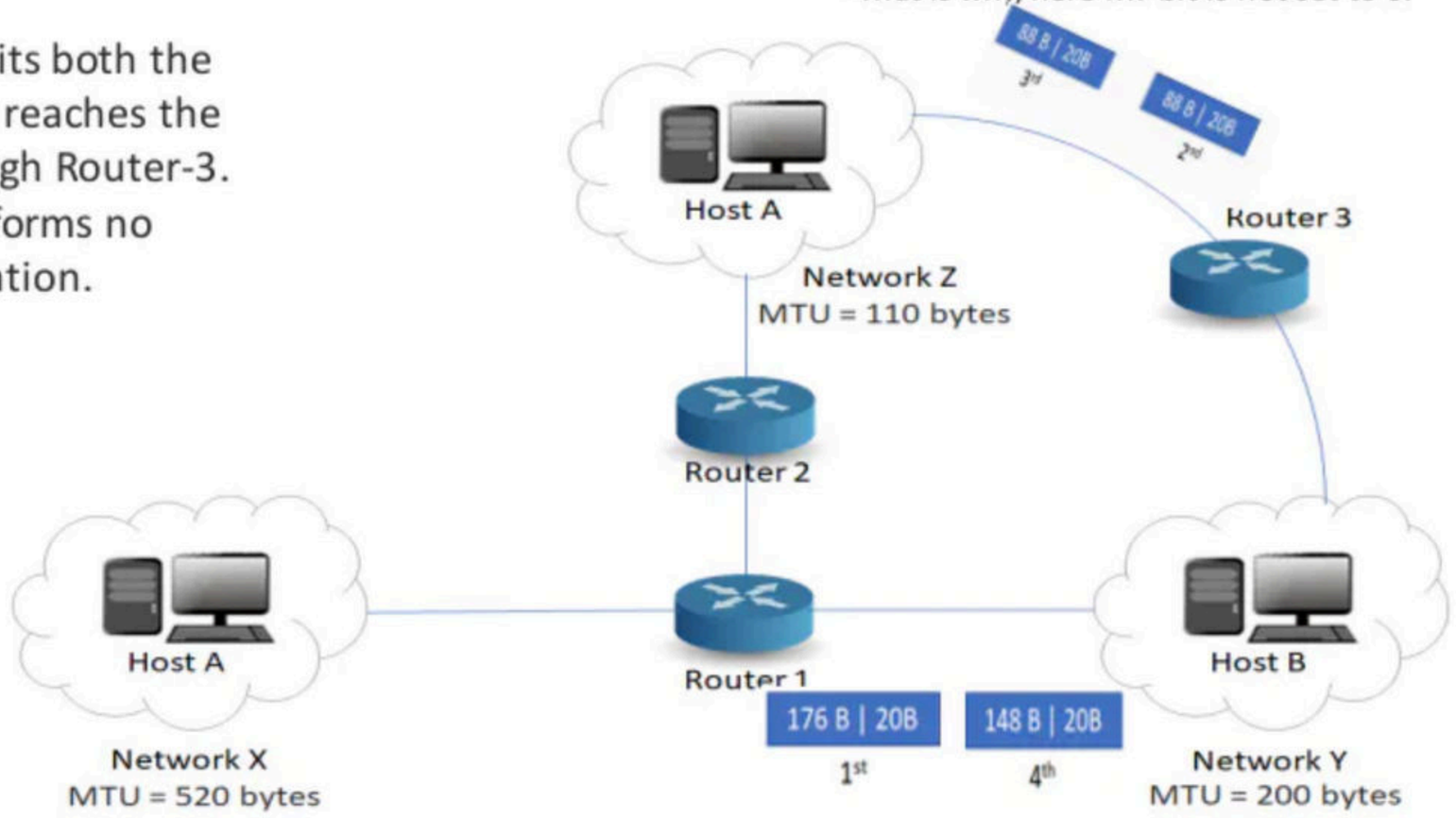
Header Information Of 2nd Fragment-

- Header length field value = 20 / 4 = 5
- Total length field value = 88 + 20 = 108
- MF bit = 1
- Fragment offset field value = (176 + 88) / 8 = 33
- Header checksum is recalculated.
- Identification number is same as that of original datagram.

Router-2 transmits both the fragments which reaches the destination through Router-3. Router-3 performs no fragmentation.

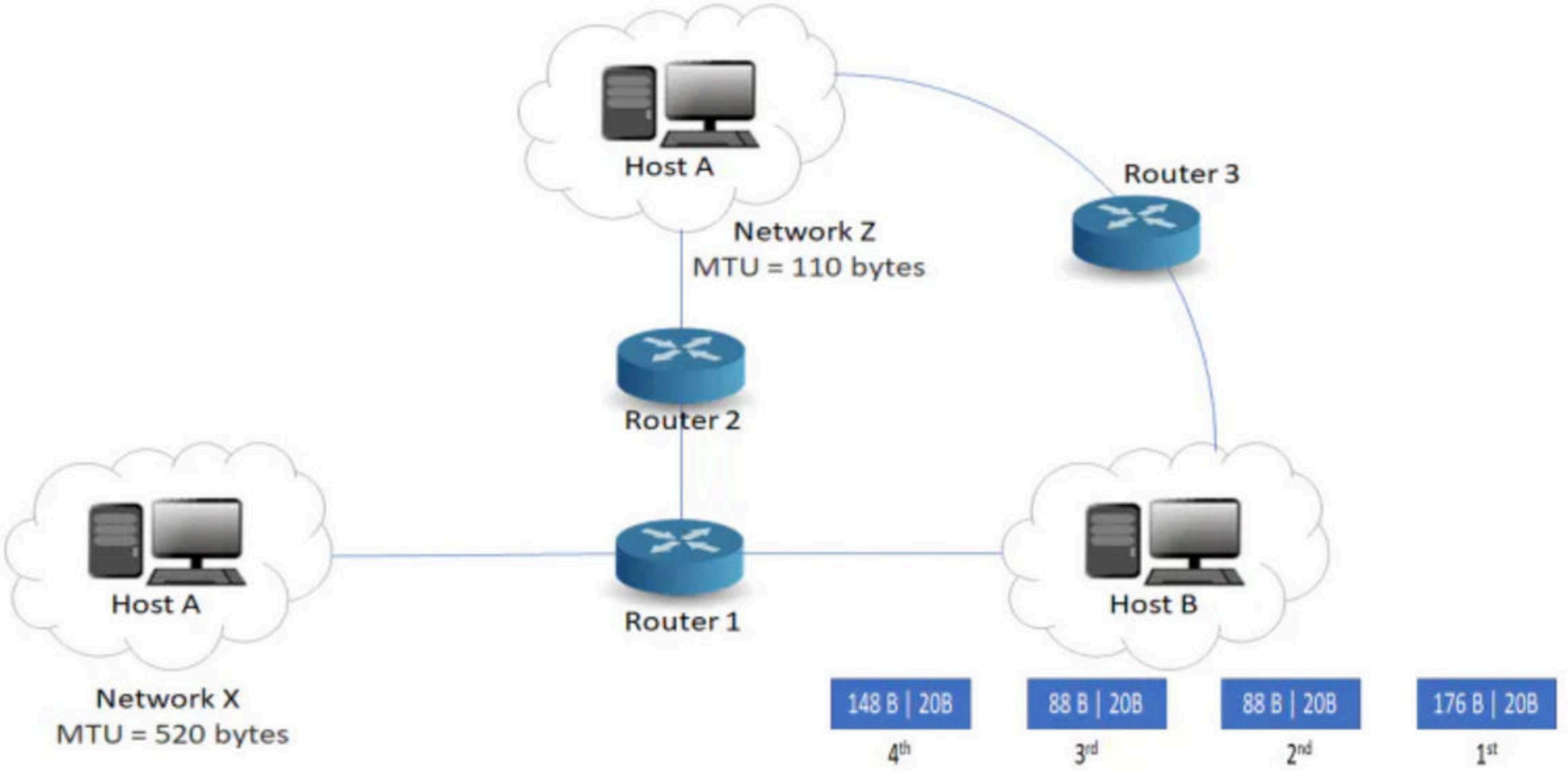
NOTE-

- This fragment is **NOT** the last fragment of the original datagram.
- It is the last fragment of the datagram received by Router-2.
- The datagram received by Router-2 is the second fragment of the original datagram.
- This datagram will serve as the third fragment of the original datagram.
- There is another fragment of the original datagram that follows it.
- That is why, here MF bit is not set to 0.



At destination side,

- Receiver receives 4 fragments of the datagram.
- Reassembly algorithm is applied to combine all the fragments to obtain the original datagram.



Fragment Offset field value for the next subsequent fragment
= (Payload length of the current fragment / 8) + Offset field value of the current fragment
= (Total length – Header length / 8) + Offset field value of the current fragment

Fragmentation Overhead

- Fragmentation of datagram increases the overhead.
- This is because after fragmentation, IP header has to be attached with each fragment.

Total Overhead
= (Total number of fragmented datagrams – 1) x size of IP header

Efficiency = Useful bytes transferred / Total bytes transferred
OR
Efficiency = Data without header / Data with header

Bandwidth Utilization or Throughput = Efficiency x Bandwidth

NOTE:

MF bit	Offset value	Represents
1	0	1st Fragment
1	!=0	Intermediate Fragment
0	!=0	Last Fragment
0	0	No Fragmentation

Reassembly is not done at the routers because-

All the fragments may not meet at the router.

Fragmented datagrams may reach the destination through independent paths.

There may be a need for further fragmentation.