IPV4

Complete Course on Computer Networks - Part III

Ravindrababu Ravula • Lesson 1 • Mar 23, 2021

23-03-2021 Unacademy Special Classes by Ravindrababu Ravula

Lecture Name	Time
Actions upo Constraint Violations, Entity & Referential Integrity Constraints	6:00 AM
Computer Networks | IPV4	7:00 AM
What makes Java Different? | L:2 | Getting Started with Java! | Ravindrababu Ravula	8:05 AM
GATE Practice Questions on OSI and TCP IP Protocol Stacks	9:10 AM
GATE Practice Questions on Multiplexers	10:15 AM
OS | lecture 35 | Paging | Part 7 (Unacademy)	11:20 AM
Practice Questions on LAN Technologies (Ethernet)	12:30 AM
Practice Questions on Analysis of Algorithms Part-1	02:00 PM
Conditional statements in Java | LIVE coding	06:00 PM
for_each in C++ STL	07:30 PM
max, max_element, min, min_element in C++ STL	08:15 PM
adjacent_difference, adjacent_find in C++ STL	09:00 PM
Stack in C++ STL	09:45 PM
Party and Handshakes Puzzle	07:45 PM
Celebrity Puzzle	08:30 PM

unacademy

RBR PLUS TEST SERIES

Test	Description	#Questions	Time	Marks	Date (18:00 hrs IST on the day of exam)
CN Topic Test – 1	Classification of addresses, Subnetting, Supernetting	15	45min	25	2 Apr 2021
CN Topic Test – 2	ISO/OSI stack, applications of each layer, bridges, routers, gateways, Flow Control Methods - Stop wait, Go back N, SR; Access control - TDM, Polling; CSMA/CD, Exponential Back off algorithm, Transmission, propagation delays etc, CRC, Aloha technologies, Framing, bit stuffing, ethernet, NAT	15	45min	25	5 Apr 2021
CN Topic Test – 3	TCP, error control algorithms, Congestion control algorithms, Switching networks, IP, fragmentation and reassembly	15	45min	25	8 Apr 2021
CN Topic Test – 4	Routing Algorithms, Application layer protocols - DNS, HTTP, FTP, SMTP, POP;	15	45min	25	11 Apr 2021
CN Subject Test	Computer Networks Complete Syllabus	33	90min	50	14 Apr 2021
OS Topic Test – 1	Processes - States, PCB, Schedulers, Threads and System calls, CPU scheduling - FCFS, SJFS, priority scheduling, Round robin, multi level queue scheduling, multilevel feedback queue scheduling	15	45min	25	17 Apr 2021
OS Topic Test – 2	Process synchronization - race conditions, critical sections, peterson's solution, Test and set locks, Semaphores, Dead locks, Starvation, Priority Inversion	15	45min	25	20 Apr 2021

OS Topic Test – 3	Memory Management - address binding, logical vs physical addressing, Dynamic loading and linking, swapping, memory allocation, fragmentation, segmentation and paging	15	45min	25	23 Apr 2021
OS Topic Test – 4	Virtual memory - Demand paging, page replacement algorithms, Thrashing, File Systems	15	45min	25	26 Apr 2021
OS Subject Test	Operating Systems Complete Syllabus	33	90min	50	29 Apr 2021
DB Topic Test – 1	ER Model and Relational Database Model, Conversion of ER Model to Relational data base model, Normalisation I (Intro to FD's , Determining candidate keys, Equivalence of FD's, FD preserving, lossless decomposition, 1NF, 2NF, 3NF, BCNF)	15	45min	25	2 May 2021
DB Topic Test – 2	Relational Algebra I (Selection, Projection, Renaming, Joins, Division, set operations, Cartesian product, Tuple Relational Calculus, Domain Relational Calculus)	15	45min	25	5 May 2021
DB Topic Test – 3	SQL	15	45min	25	8 May 2021
DB Topic Test – 4	Transactions, Serializability, Locking - 2PL, Timestamp protocol, Thomas write rule, Graph based protocol. File Structures - Indexing, B and B+ trees - insertion, deletion.	15	45min	25	11 May 2021

DB Subject Test	Databases Complete Syllabus	33	90min	50	14 May 2021
DLD Topic Test – 1	Logic functions, Minimization	15	45min	25	17 May 2021
DLD Topic Test – 2	Design and Synthesis of Combinational circuits	15	45min	25	20 May 2021
DLD Topic Test – 3	Sequential Circuits, Number Systems	15	45min	25	23 May 2021
DLD Subject Test	Digital Logic Complete Syllabus	33	90min	50	26 May 2021
P & DS Topic Test – 1	Arrays - 1D, 2D, multidimensional, row and column major orders, C programming: Functions, recursion, scope of variables	15	45min	25	29 May 2021
P & DS Topic Test – 2	Stacks, Queues, prefix-postfix conversion, towers of hanoi, postfix evaluation	15	45min	25	1 Jun 2021
P & DS Topic Test – 3	Pointers, function pointers, Structures, Unions, Linked list	15	45min	25	4 Jun 2021
P & DS Topic Test – 4	Binary Heaps, Graphs, AVL trees, Binary search trees	15	45min	25	7 Jun 2021
P & DS Subject Test	Programming and Data Structures Complete Syllabus	33	90min	50	10 Jun 2021
Algorithms Topic Test – 1	Asymptotic worst case time and space complexity	15	45min	25	13 Jun 2021

Algorithms Topic Test – 2	Searching, Hashing, Sorting, Divide & Conquer,	15	45min	25	16 Jun 2021
Algorithms Topic Test – 3	Greedy Algorithms	15	45min	25	19 Jun 2021
Algorithms Topic Test – 4	Dynamic programming, Graph Search	15	60min	25	22 Jun 2021
Algorithms Subject Test	Algorithms Complete Syllabus	33	90min	50	25 Jun 2021
TOC Topic Test – 1	DFA, NFA, conversion of NFA to DFA, Moore and Mealey machines	15	45min	25	28 Jun 2021
TOC Topic Test – 2	Regular Expressions and Conversions, Finding whether a given grammar is Regular, Pumping Lemma	15	45min	25	1 Jul 2021
TOC Topic Test – 3	Grammars and Chomsky hierarchy, CNF and GNF, Context free languages and push down automata, pumping lemma for CFLs, Turing machines, context sensitive languages,	15	45min	25	4 Jul 2021
TOC Topic Test – 4	Recursive and recursively enumerable languages, Closure properties of all languages, Undecidability	15	45min	25	7 Jul 2021
TOC Subject Test	Theory of Computation	33	90min	50	10 Jul 2021
CD Topic Test – 1	Stages in compilers, ambiguity, Lexical Analysis, Parsing - I (Recursive Descent, operator precedence, LL(1))	15	45min	25	13 Jul 2021

CD Topic Test – 2	Parsing - II (LR(0), SLR, CLR, LALR)	15	45min	25	16 Jul 2021
CD Topic Test – 3	Syntax Directed Translation, Intermediate Code Generation, Run Time Environment, DAG, Liveness Analysis, Common sub expression elimination	15	45min	25	19 Jul 2021
CD Subject Test	Compiler Design	33	90min	50	22 Jul 2021
EM Topic Test – 1	Propositional and first order logic; Sets, Relations, Functions, Partial Order, Lattices and Groups	15	45min	25	25 Jul 2021
EM Topic Test – 2	Graphs: Connectivity, Matching, Coloring, Combinatorics: Counting, Recurrance Relations, Generating Functions	15	45min	25	28 Jul 2021
EM Topic Test – 3	Linear Algebra: Matrices , Determinants, System of Linear Equations, Eigen values and Eigen vectors, LU Decomposition	15	45min	25	31 Jul 2021
EM Topic Test – 4	Calculus: Limits, Continuity and Differentiability, Maxima and Minima, Mean Value Theorem, Integration	15	45min	25	3 Aug 2021

EM Topic Test – 5	Probability: Random Variables, Uniform, Normal, Exponential, Poisson and Binomial Distributions, Conditional probability, Bayes theorem. Mean, Median, Mode and Standard Deviation.	15	45min	25	6 Aug 2021
EM Subject Test	Engineering Mathematics Complete Syllabus	33	90min	50	9 Aug 2021
CO Topic Test – 1	Memory Interfacing, Hierarchy, Cache block replacement policies, Cache Mapping techniques - Direct mapping, associative, and set associative mapping	15	45min	25	12 Aug 2021
CO Topic Test – 2	Machine Instructions and Addressing Modes, ALU, data path, control unit - Instruction cycle, interrupt driven IO, Hardwired control Unit, Pipelining	15	45min	25	15 Aug 2021
CO Topic Test – 3	Secondary Memory, IO Interface, Micro programmed Control Unit	15	45min	25	18 Aug 2021
CO Subject Test	Computer Organization Complete Syllabus	33	90min	50	21 Aug 2021

Aptitude	Aptitude Test – 1(Averages, Ages, Ratio and proportion, Profit and Loss, Simple Interest, Compound Interest , Alligation and Mixture, Partnership, Time and Work)	15	45min	25	24 Aug 2021
	Aptitude Test – 2(Mensuration, LCM and HCF, Permutations and Combinations, Time and distance)	15	45min	25	27 Aug 2021
	Aptitude Test – 3(Data Sufficiency, Equations, Introduction to Data-Interpretation, Logarithms, Progression, Surds and Indices, Number Systems, Alphabetic series, Number series.)	15	45min	25	30 Aug 2021
	Aptitude Test – 4(Blood Relation, Directions and Distance, Circular Arrangements Linear Arrangements, Coded Inequalities, Alphabet Test, Element Series, Ranking (or) Ordering, Coding and Decoding, Percentages)	15	45min	25	2 Sep 2021
	Aptitude Test Spatial – 5	15	45min	25	5 Sep 2021
	Aptitude Test Spatial– 6	15	45min	25	8 Sep 2021
	Aptitude Test	33	90min	50	11 Sep 2021

Test	Subjects	#Questions	Time	Marks	Date (18:00 hrs IST on the day of exam)
Multi Subject Test – 1	Theory of Computation, Compiler Design	33	90min	50	14 Sep 2021
Multi Subject Test – 2	Operating Systems, Databases	33	90min	50	17 Sep 2021
Multi Subject Test – 3	Programming and Data Structures, Algorithms	33	90min	50	21 Sep 2021
Multi Subject Test – 4	Digital Logic, Computer Organization	33	90min	50	25 Sep 2021
Multi Subject Test – 5	Computer Networks, Engineering Mathematics	33	90min	50	29 Sep 2021
Multi Subject Test – 6	Theory of Computation, Compiler Design, Operating Systems, Digital Logic	33	90min	50	3 Oct 2021
Multi Subject Test – 7	Databases, Programming and Data Structures, Algorithms	33	90min	50	7 Oct 2021

Multi Subject Test – 8	Engineering Mathematics, Computer Networks, Computer Organization	33	90min	50	11 Oct 2021
Multi Subject Test – 9	Theory of Computation, Compiler Design, Algorithms, Programming & Data Structures, Engineering Mathematics, Operating Systems	33	90min	50	15 Oct 2021
Multi Subject Test – 10	Databases, Computer Networks, Digital Logic, Computer Organization and Architecture, General Aptitude	33	90min	50	19 Oct 2021

Test Name	#Questions	Time	Marks	Date (18:00 hrs IST on the day of exam)
Grand Test – 1	65	180min	100	17 Oct 2021
Grand Test – 2	65	180min	100	24 Oct 2021
Grand Test – 3	65	180min	100	31 Oct 2021
Grand Test – 4	65	180min	100	7 Nov 2021
Grand Test – 5	65	180min	100	14 Nov 2021
Grand Test – 6	65	180min	100	21 Nov 2021
Grand Test – 7	65	180min	100	28 Nov 2021
Grand Test – 8	65	180min	100	5 Dec 2021
Grand Test – 9	65	180min	100	12 Dec 2021
Grand Test – 10	65	180min	100	19 Dec 2021

Test Name	#Questions	Time	Marks	Date (18:00 hrs IST on the day of exam)
Mock GATE Test – 1	65	180min	100	26 Dec 2021
Mock GATE Test – 2	65	180min	100	2 Jan 2022
Mock GATE Test – 3	65	180min	100	9 Jan 2022
Mock GATE Test – 4	65	180min	100	16 Jan 2022
Mock GATE Test – 5	65	180min	100	23 Jan 2022

Identification

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

DF Bit

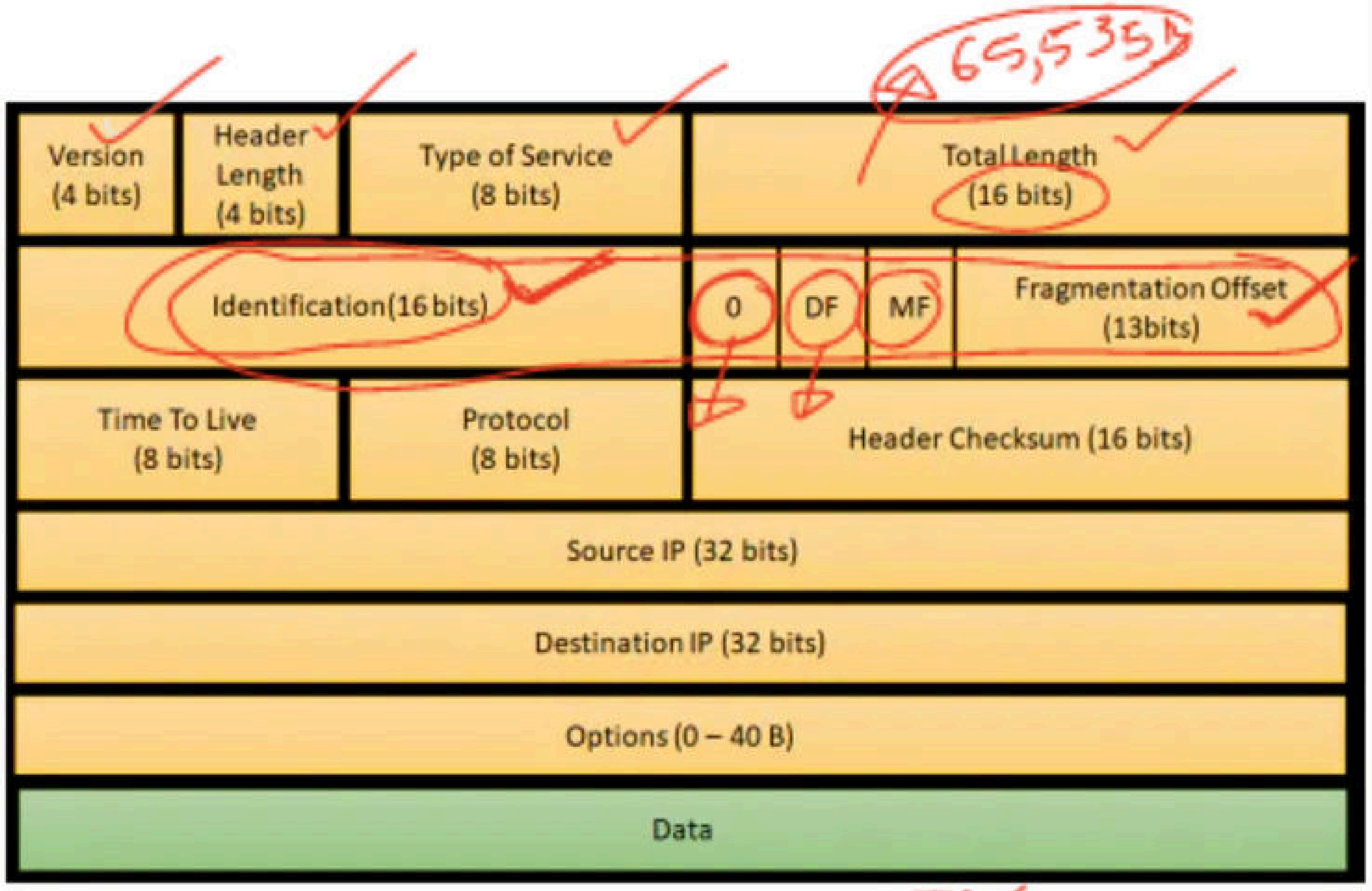
- DF bit stands for Do Not Fragment bit.
- Its value may be 0 or 1.

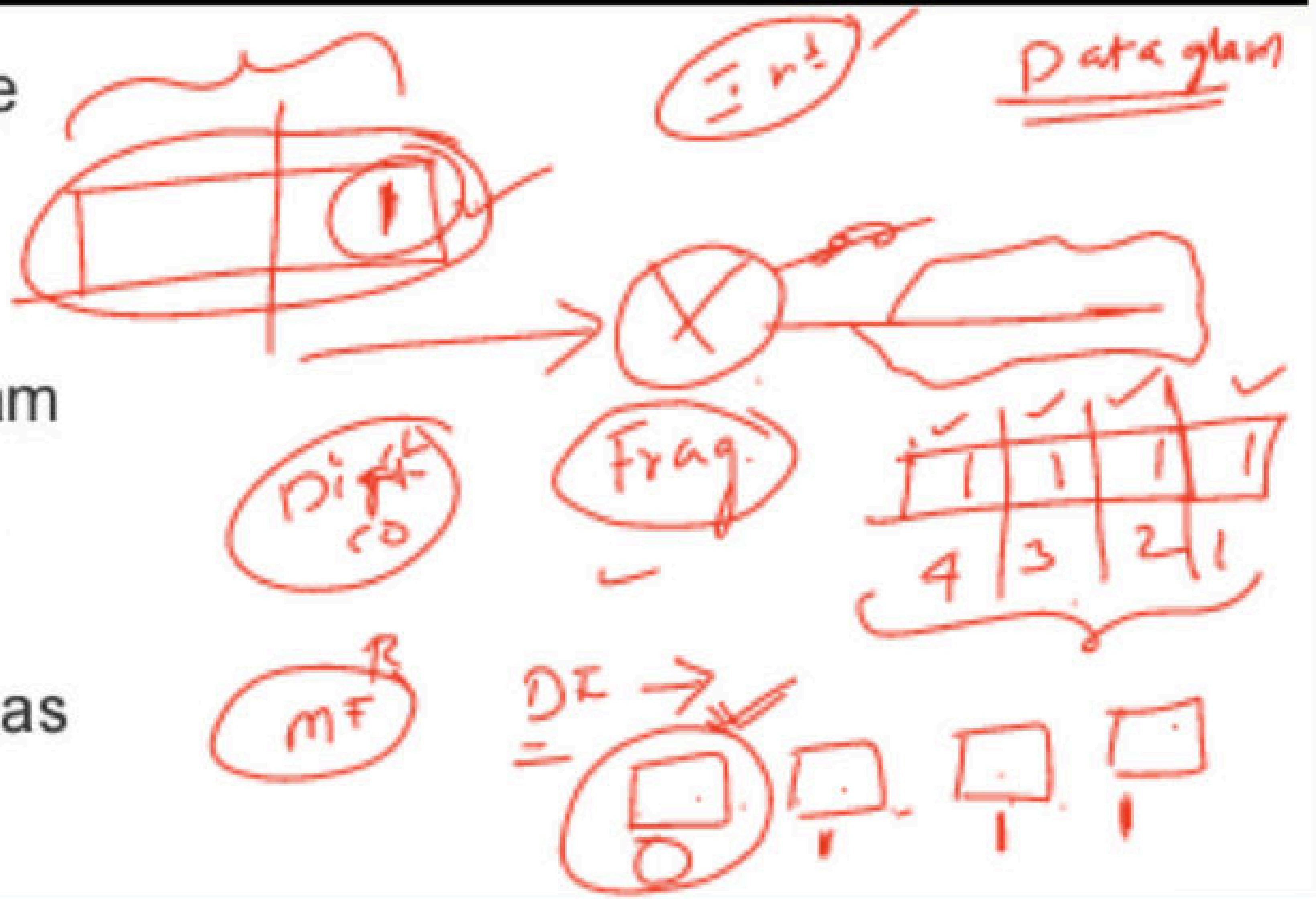
When DF bit is set to 0,

- 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.





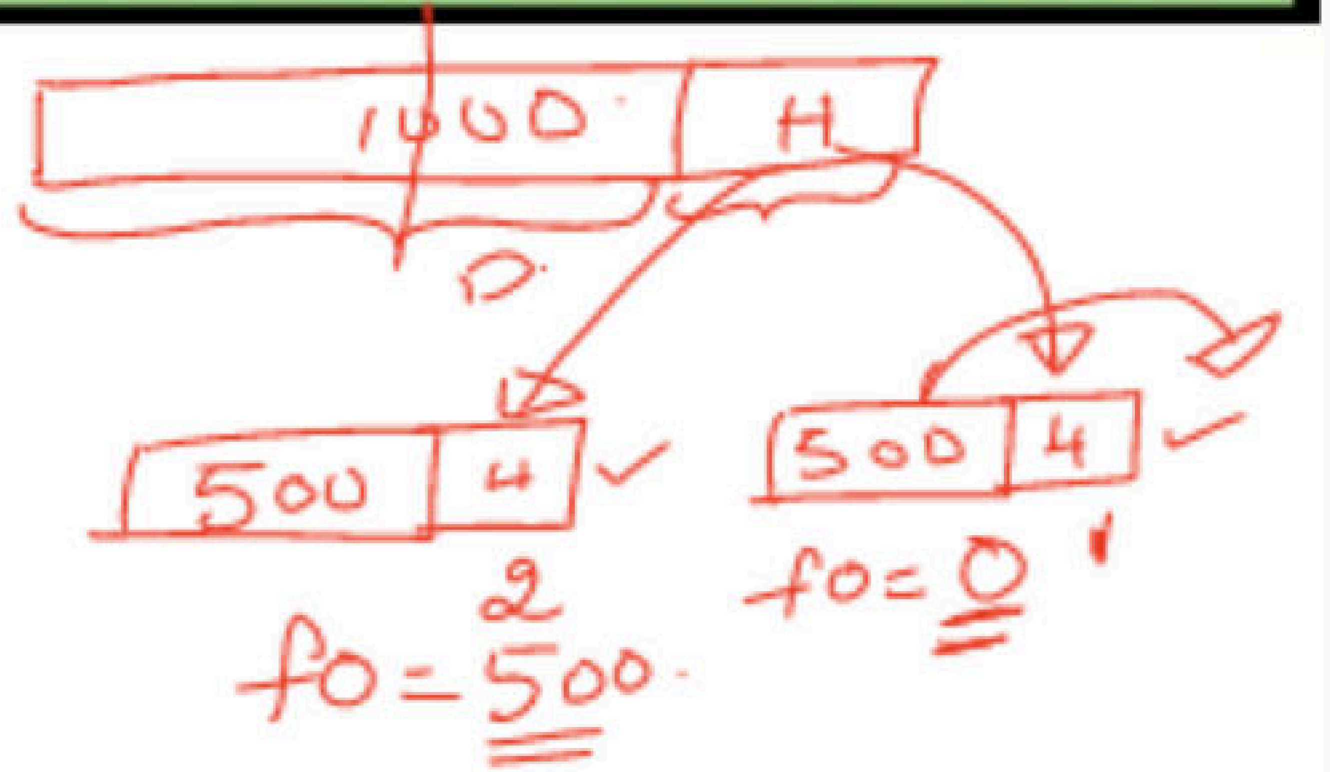
MF Bit

- 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.

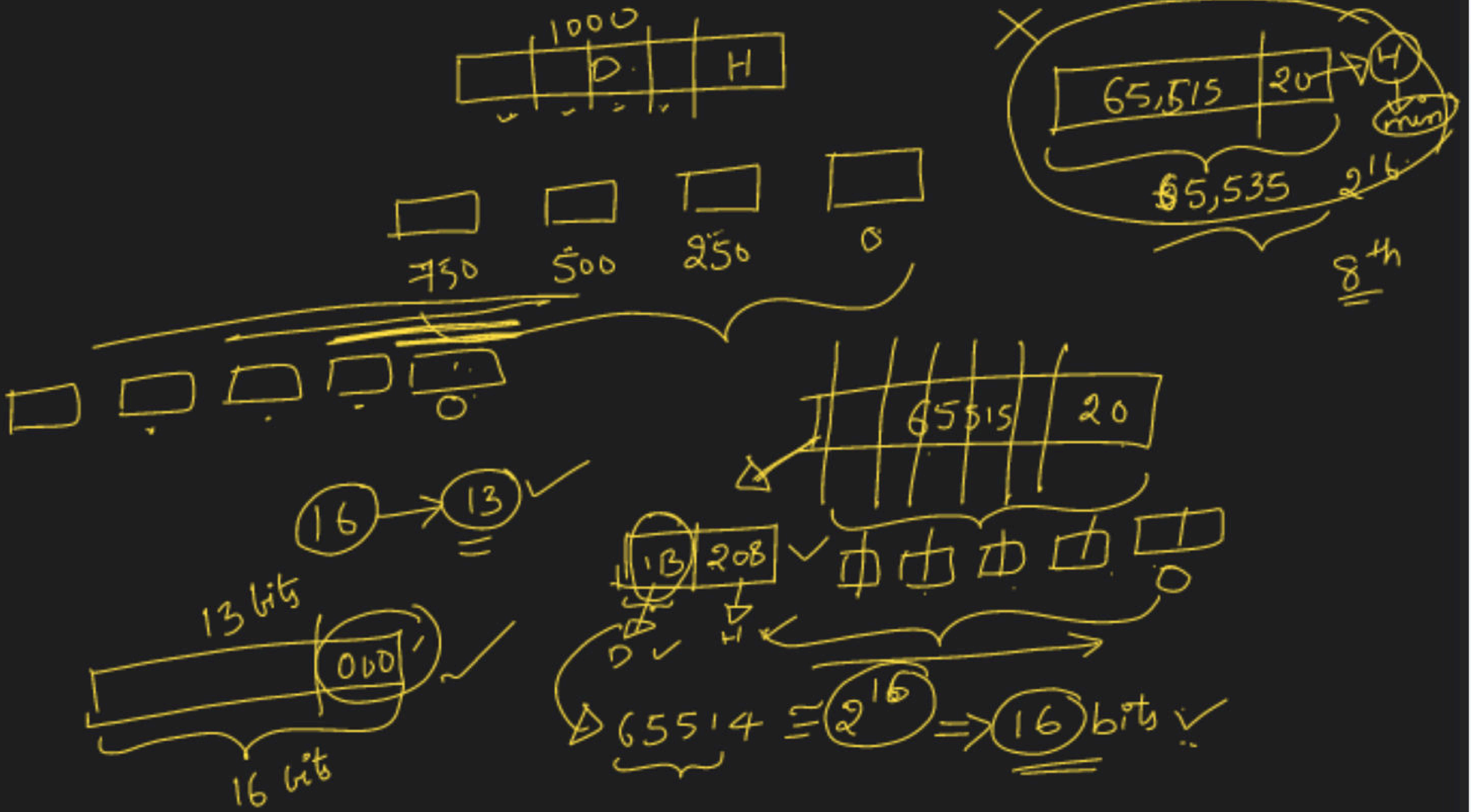
Version (4 bits)	Header Length (4 bits)	Type of Service (8 bits)	Total Length (16 bits)			
Identification(16 bits)			0	DF	MF	Fragmentation Offset (13bits)
Time To Live (8 bits)		Protocol (8 bits)	Header Checksum (16 bits)			
Source IP (32 bits)						
Destination IP (32 bits)						
Options (0 – 40 B)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



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



1000
D
H
65,515
20
H
min
65,535
2^{16}
8th
750
500
250
0
0
65515
20
16
13
1B
208
D
H
0
13 bits
000
16 bits
65514
2^{16}
16 bits

Scaling

$\rightarrow 2^{13} \leftrightarrow 2^{16}$ ✓

4 => HL

I

↓

60

bits

4

4 → ULF → *4 → HL

$\frac{2^{16}}{2^{13}} = 2^3 = 8$ ✓

=> FO

8

=> EOF

*8

18

4	3	2	1

multip

1.4	3	2	1
8X	8X	8X	0

→ 8X

18

*8

871

13 000

13 000

18

13 bits

000

16

*8

*8

S R

2^3

$*2^3$

P, P3

□ 00

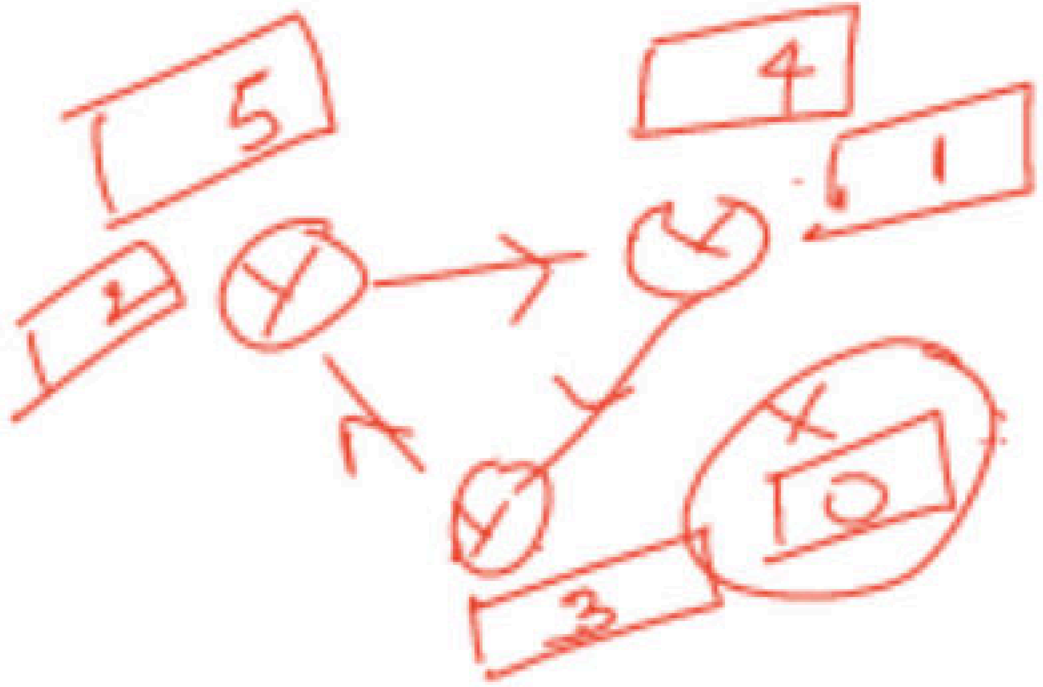
Time To Live

- 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.

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				

Time → hops

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.

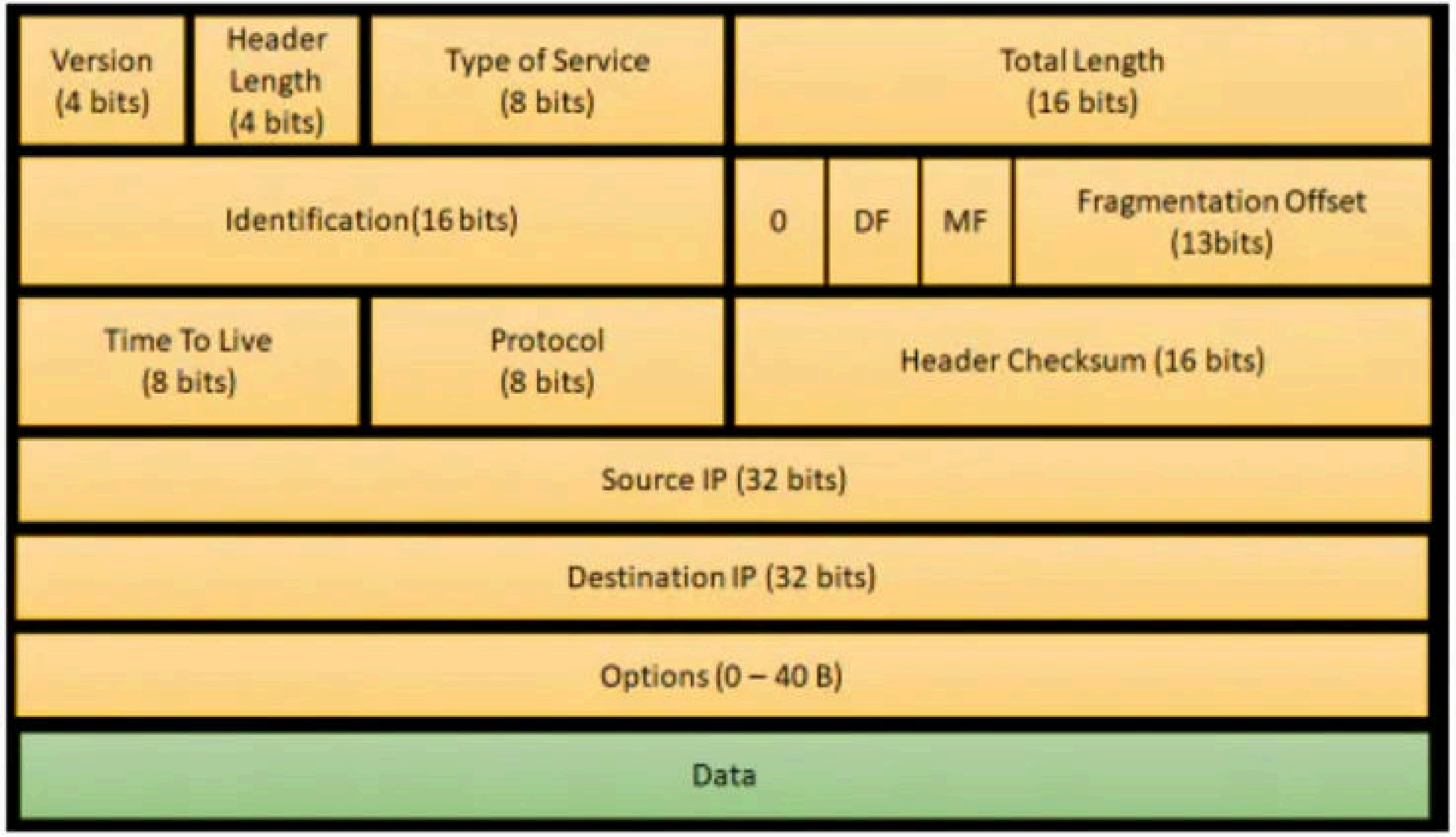


TTL

~~Soap~~ Same IP

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

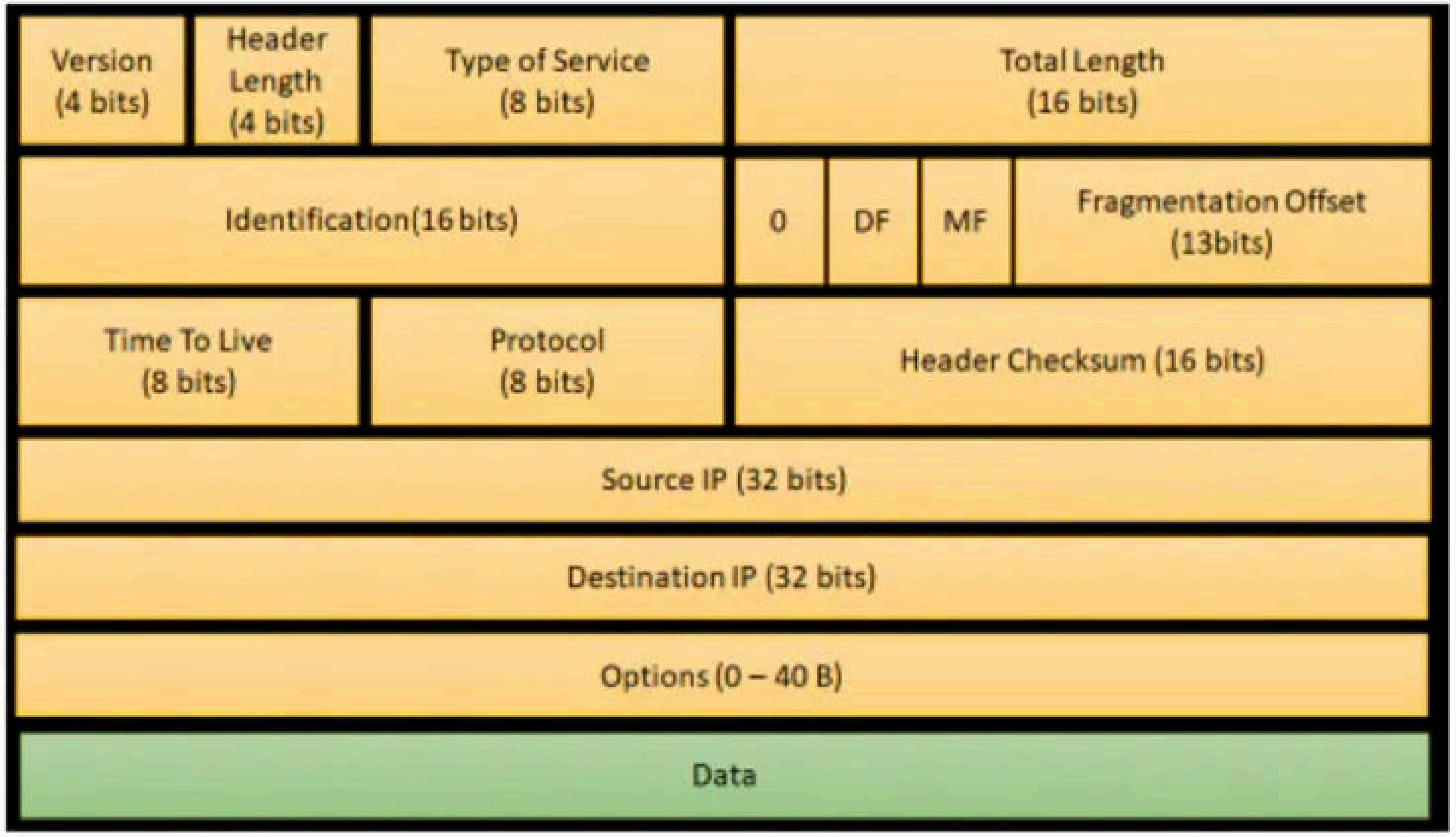
- Header checksum is a 16 bit field.
- It contains the checksum value of the entire header.
- The checksum value is used for error checking of the header.

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.

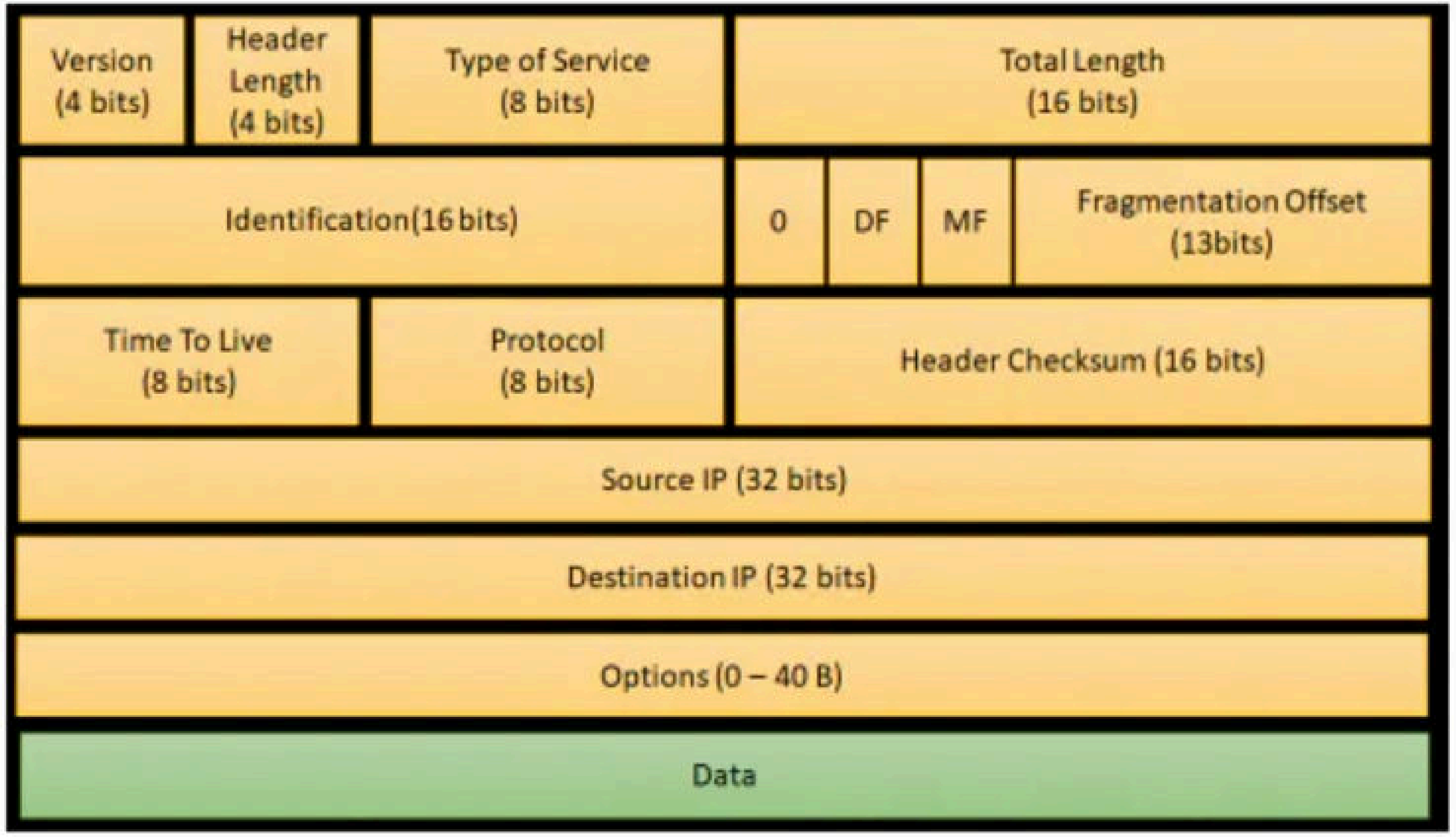
Destination IP Address

- 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.

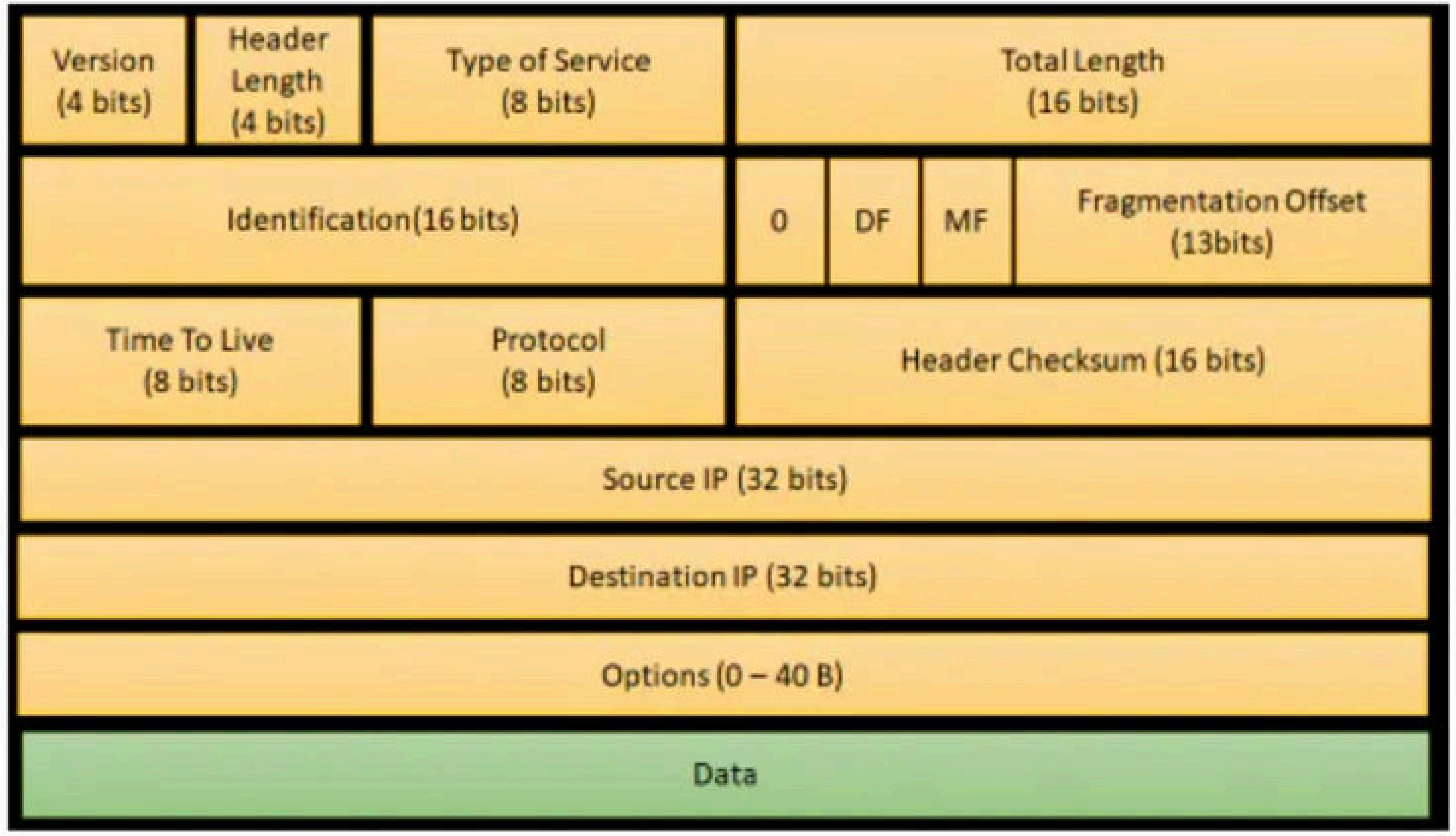
The maximum number of IPv4 router addresses that can be recorded in the Record Route option field of an IPv4 header is 9.

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.

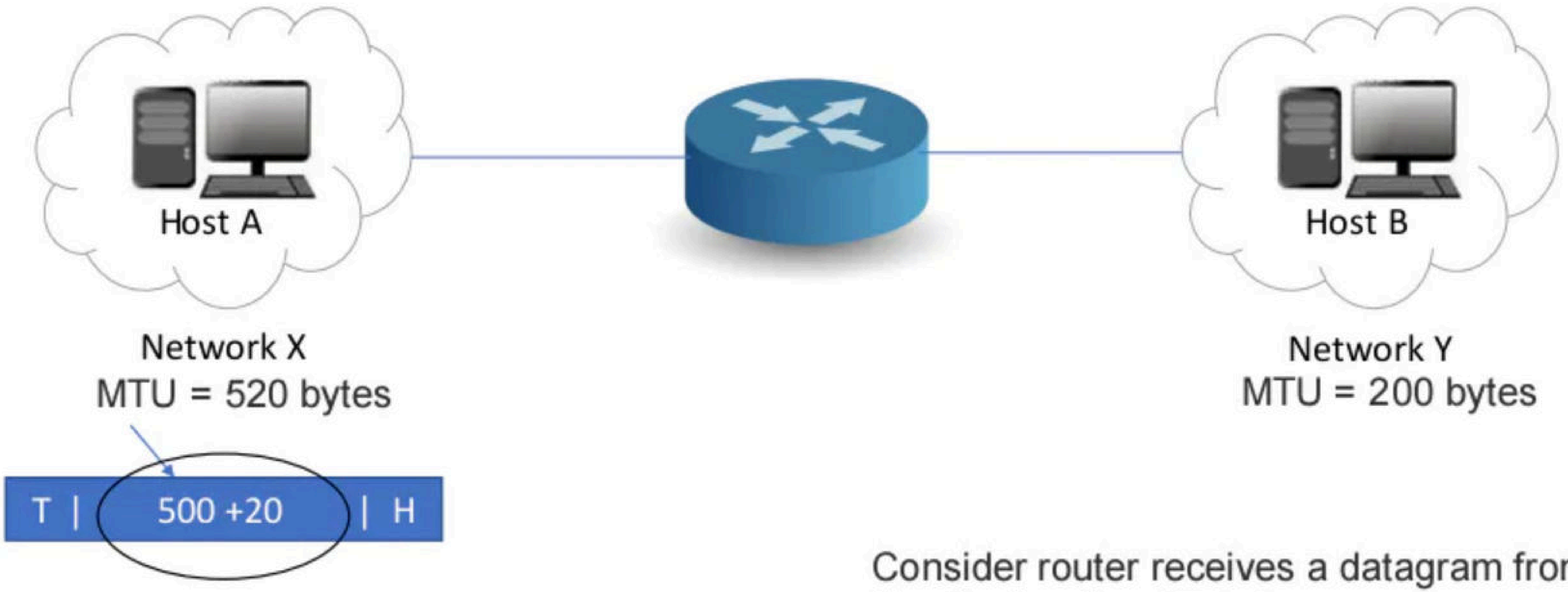
Computer Networks

Fragmentation

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

EXAMPLE 1

Host A wants to send a message to host B

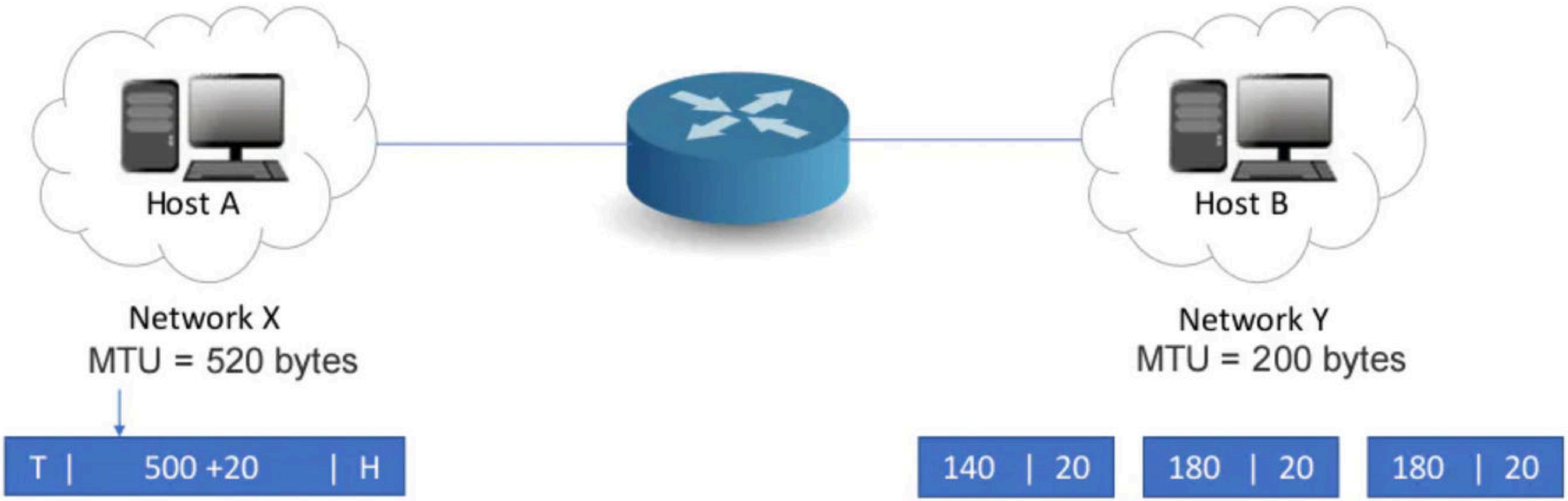


Consider router receives a datagram from host A having-
Header length = 20 bytes
Payload length = 500 bytes
Total length = 520 bytes

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

EXAMPLE 1

Host A wants to send a message to host B

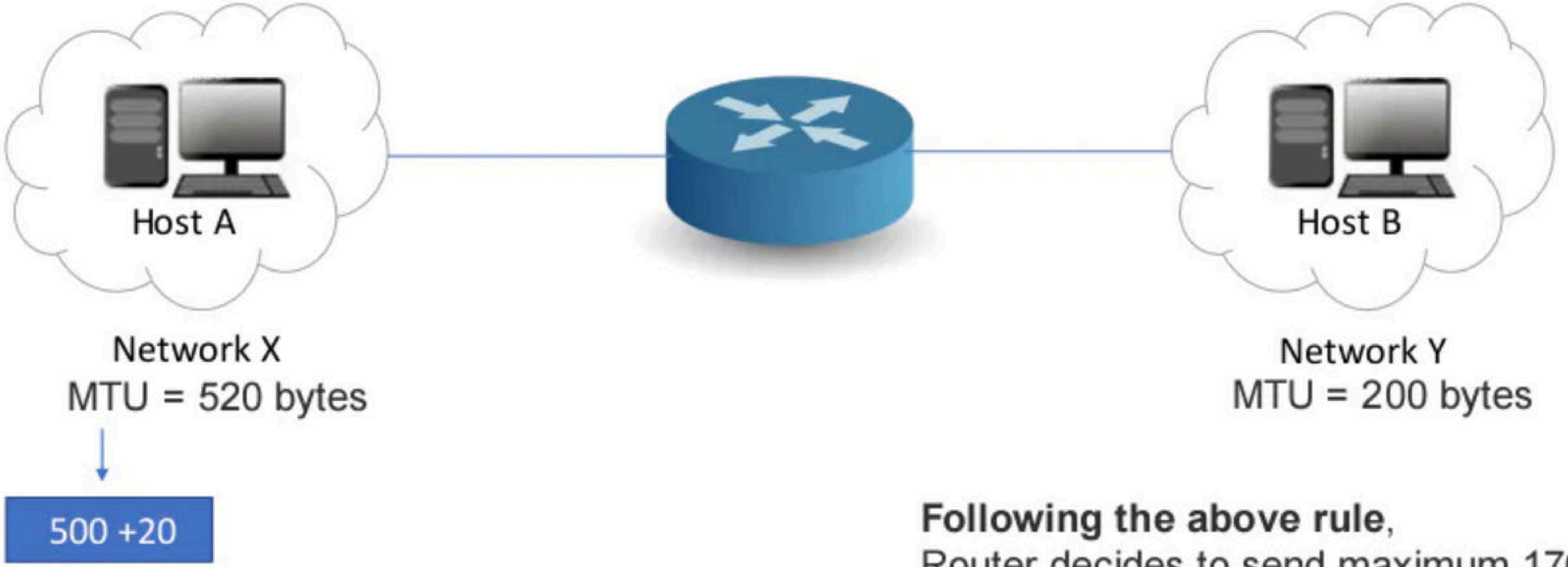


Router decides the amount of data that it should transmit in each fragment.

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.



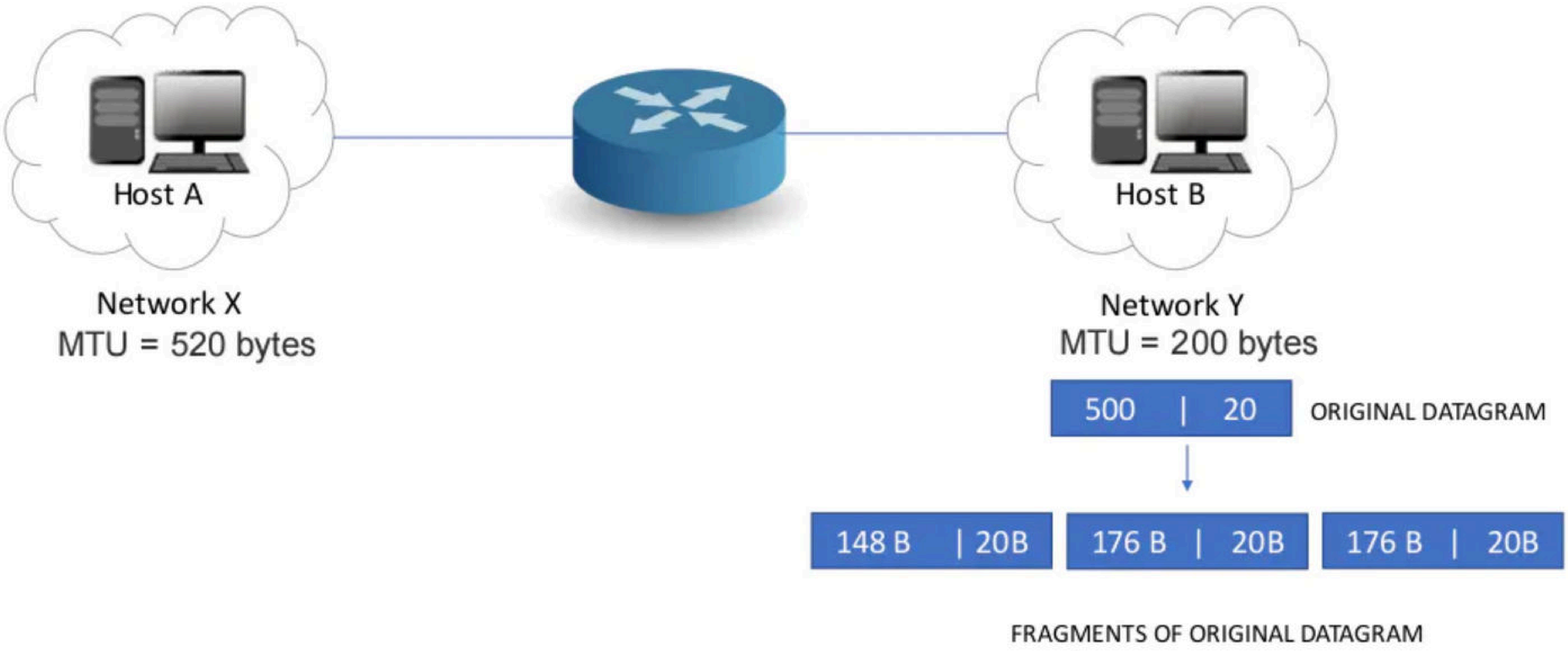
Following the above rule,
Router decides to send maximum 176 bytes of data in one fragment.
This is because it is the greatest value that is a multiple of 8 and less than MTU.

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



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.

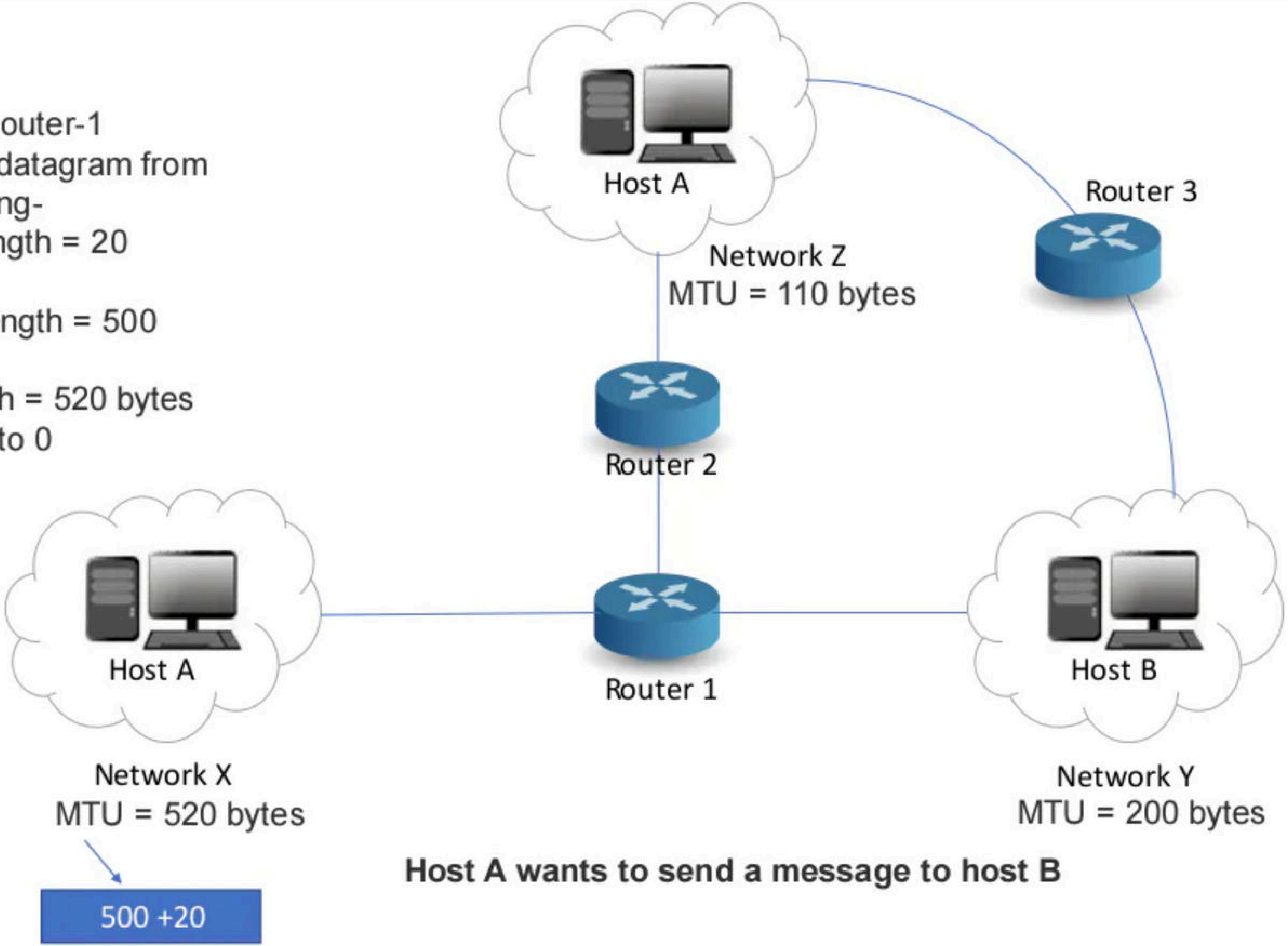
Router transmits all the fragments.

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

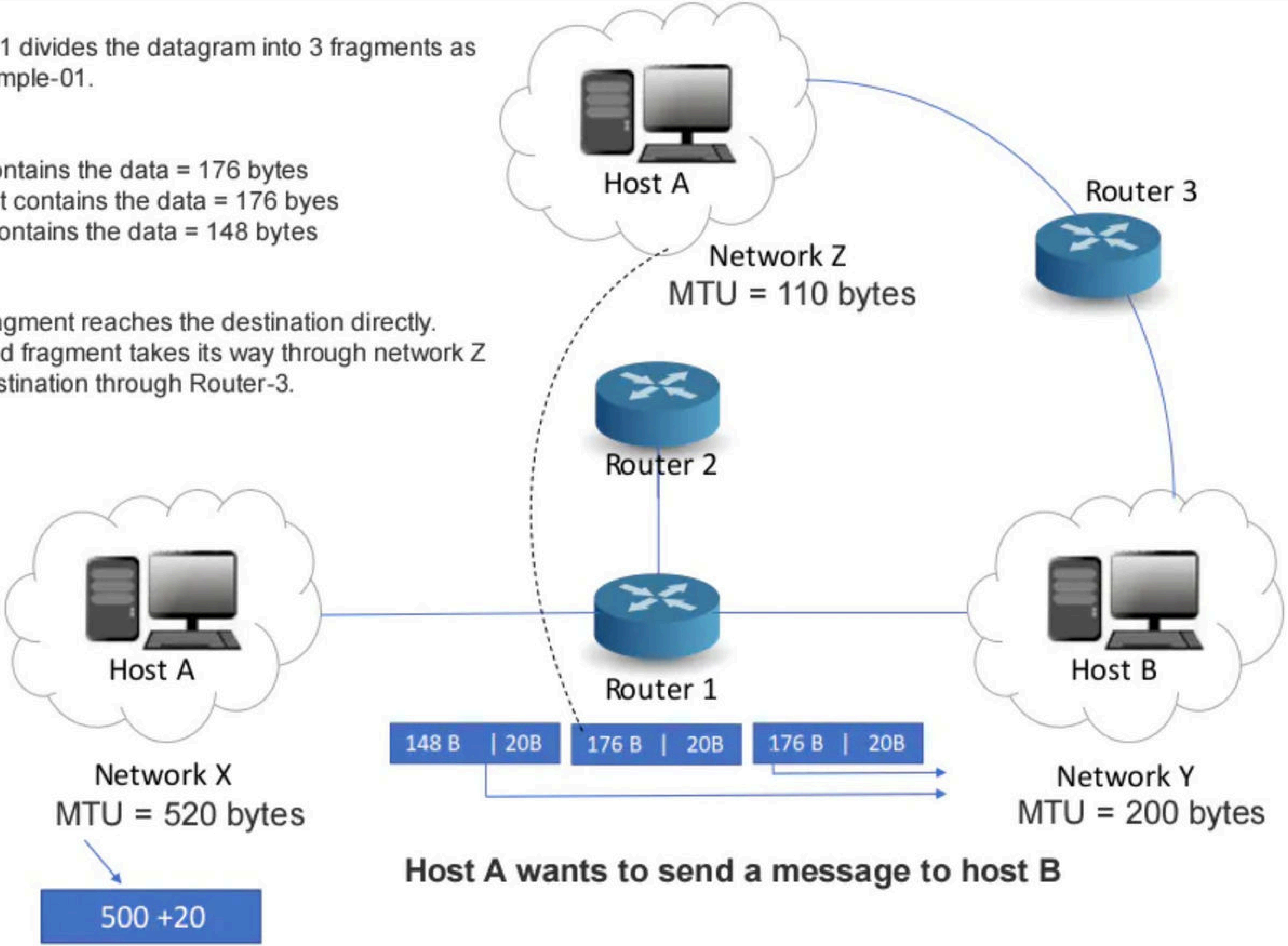
Consider Router-1 divides the datagram into 3 fragments as discussed in Example-01.

Then,

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

Now, consider-

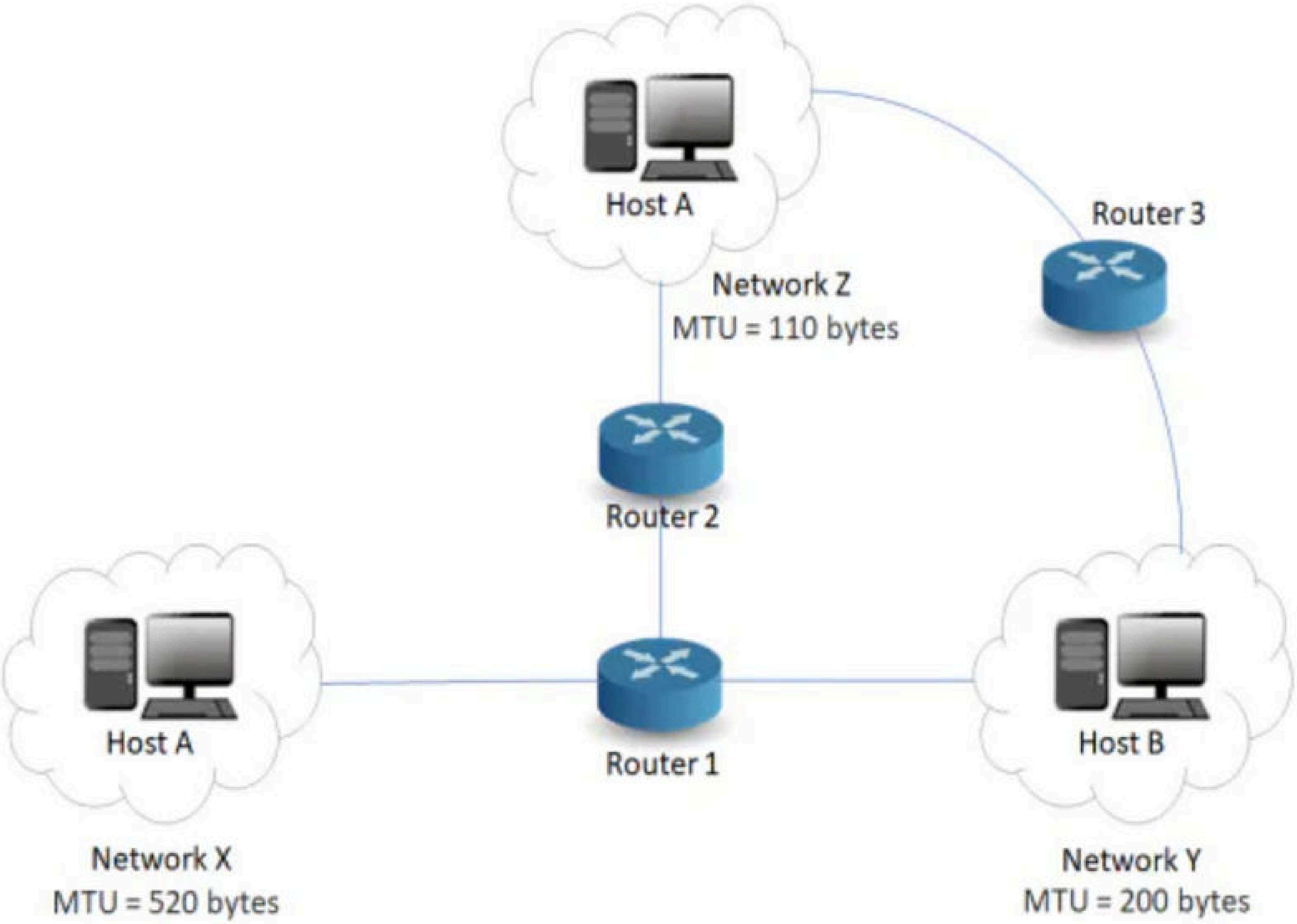
- First and third fragment reaches the destination directly.
- However, second fragment takes its way through network Z and reach the destination through Router-3.



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-

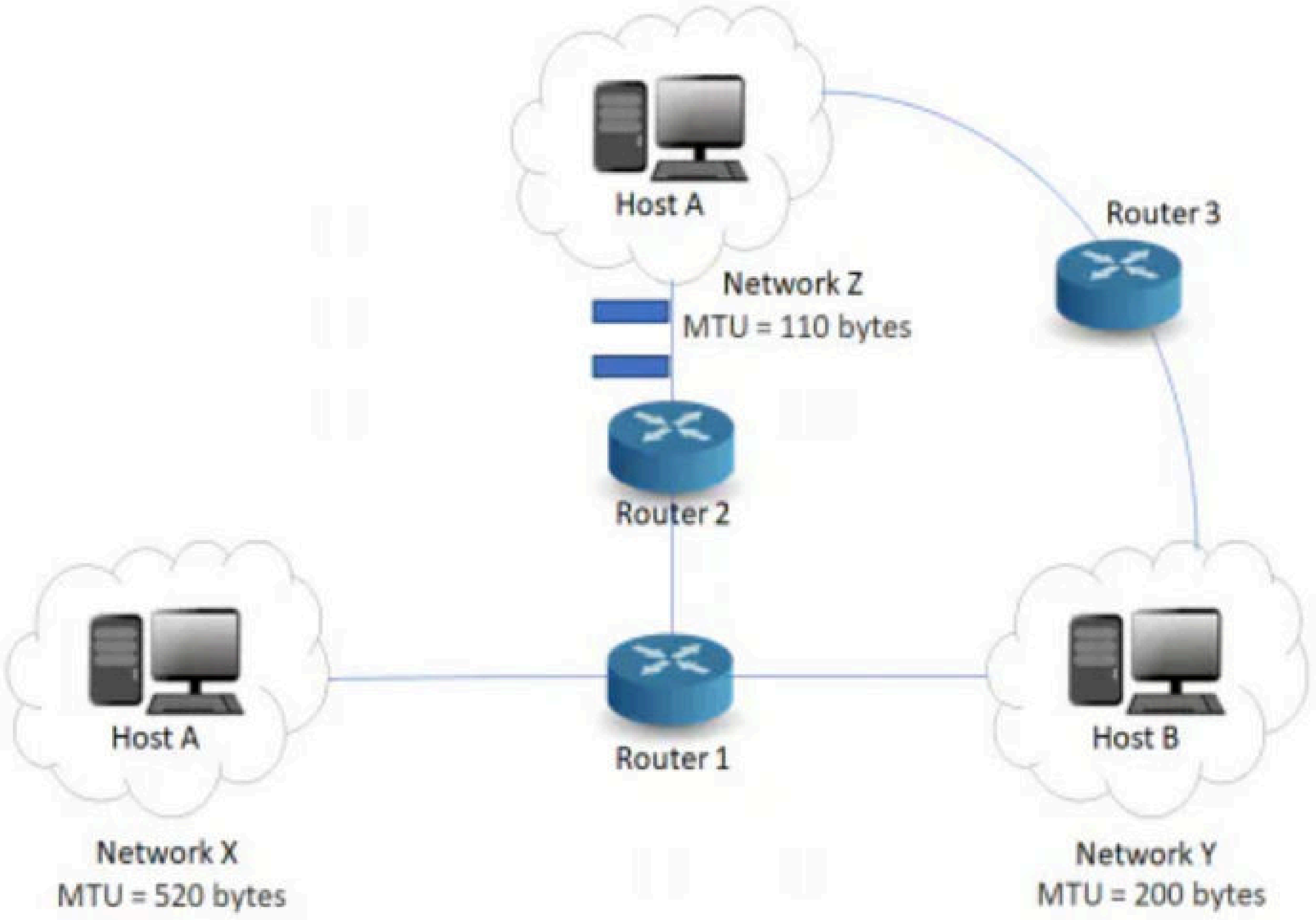
- Header length = 20 bytes
- Payload length = 176 bytes
- Total length = 196 bytes
- DF bit set to 0



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.



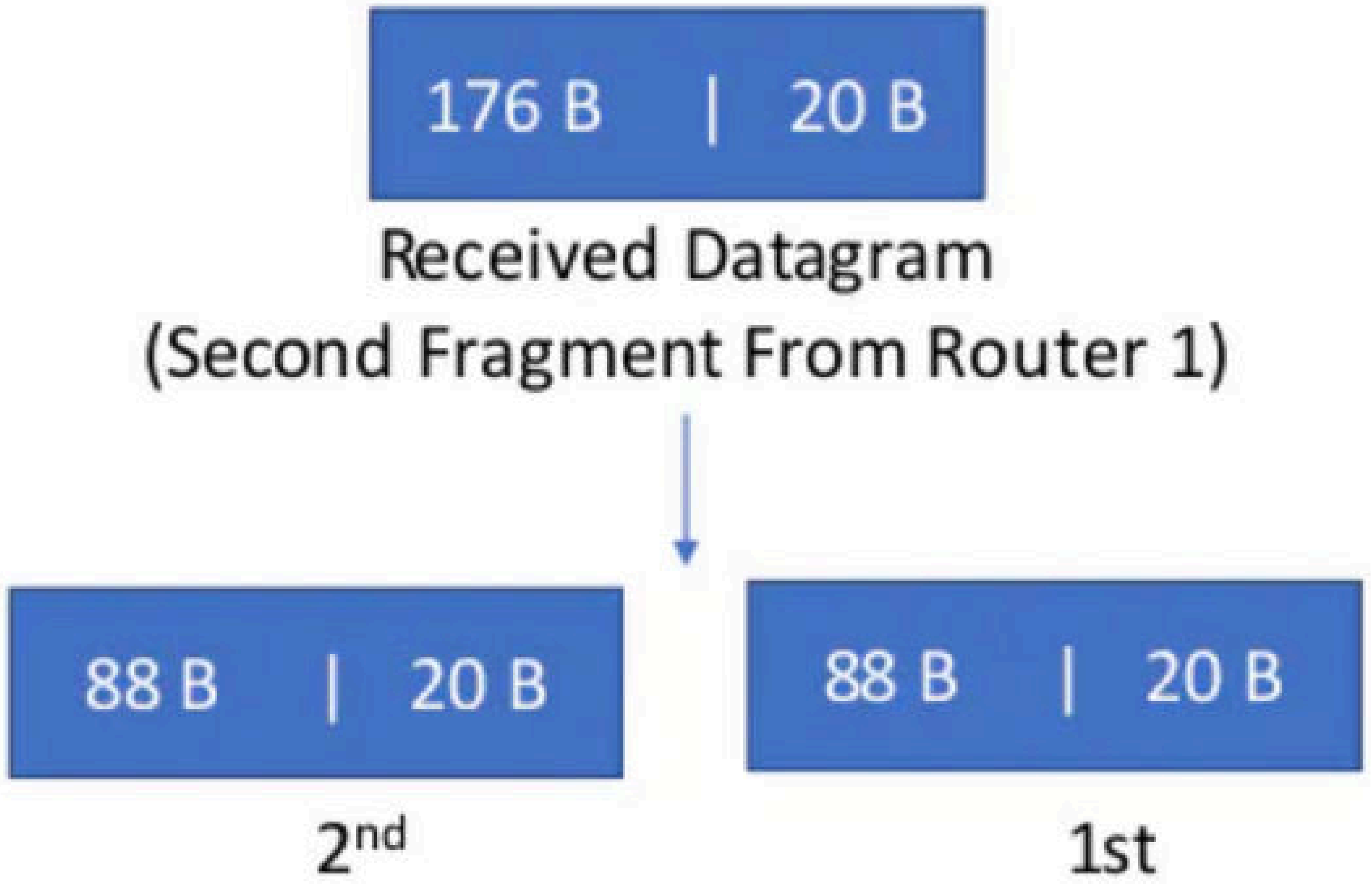
Step-02:

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 origina 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 th original datagram.

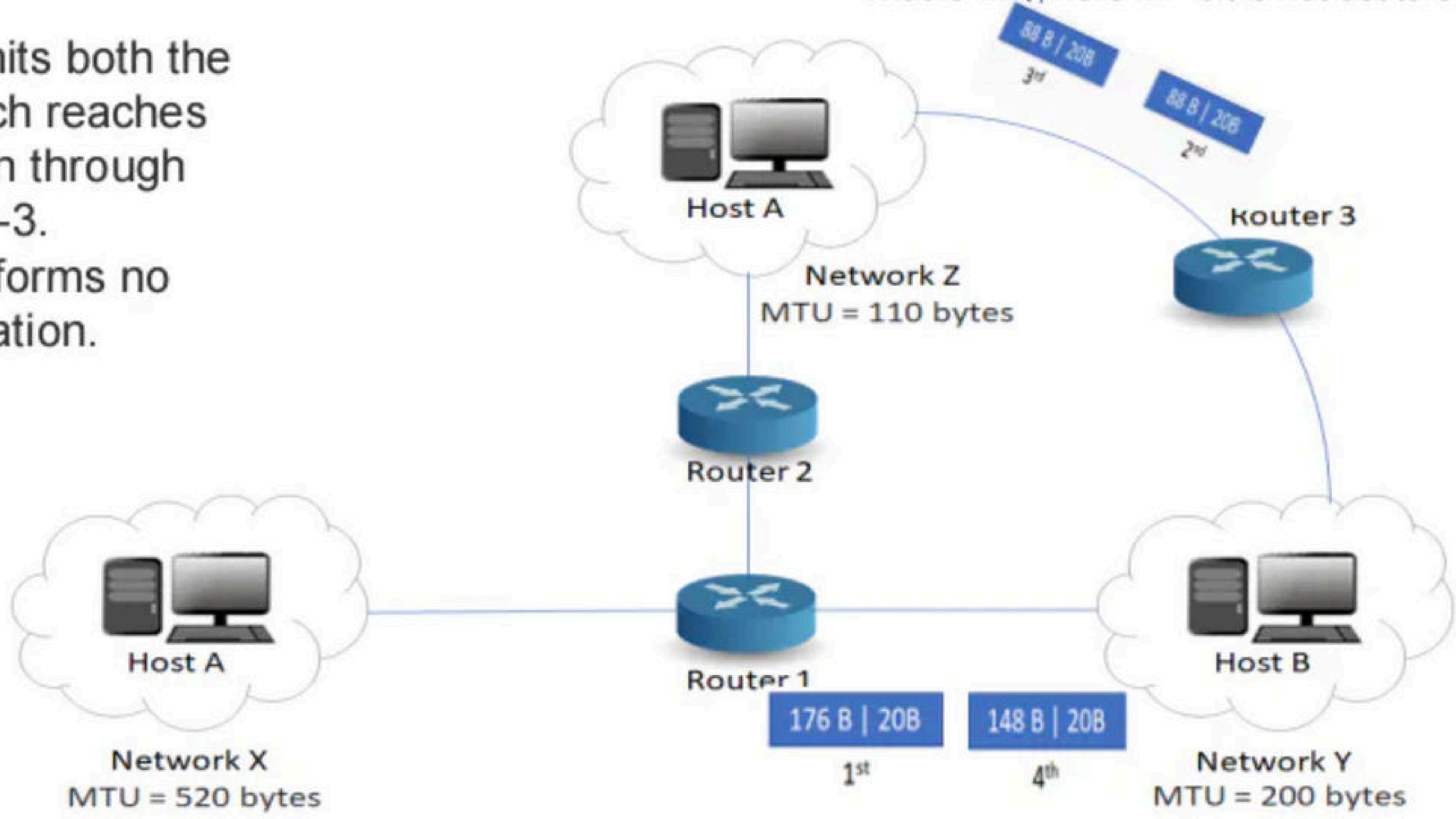
Header Information Of 2nd Fragment-

- 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.

NOTE-

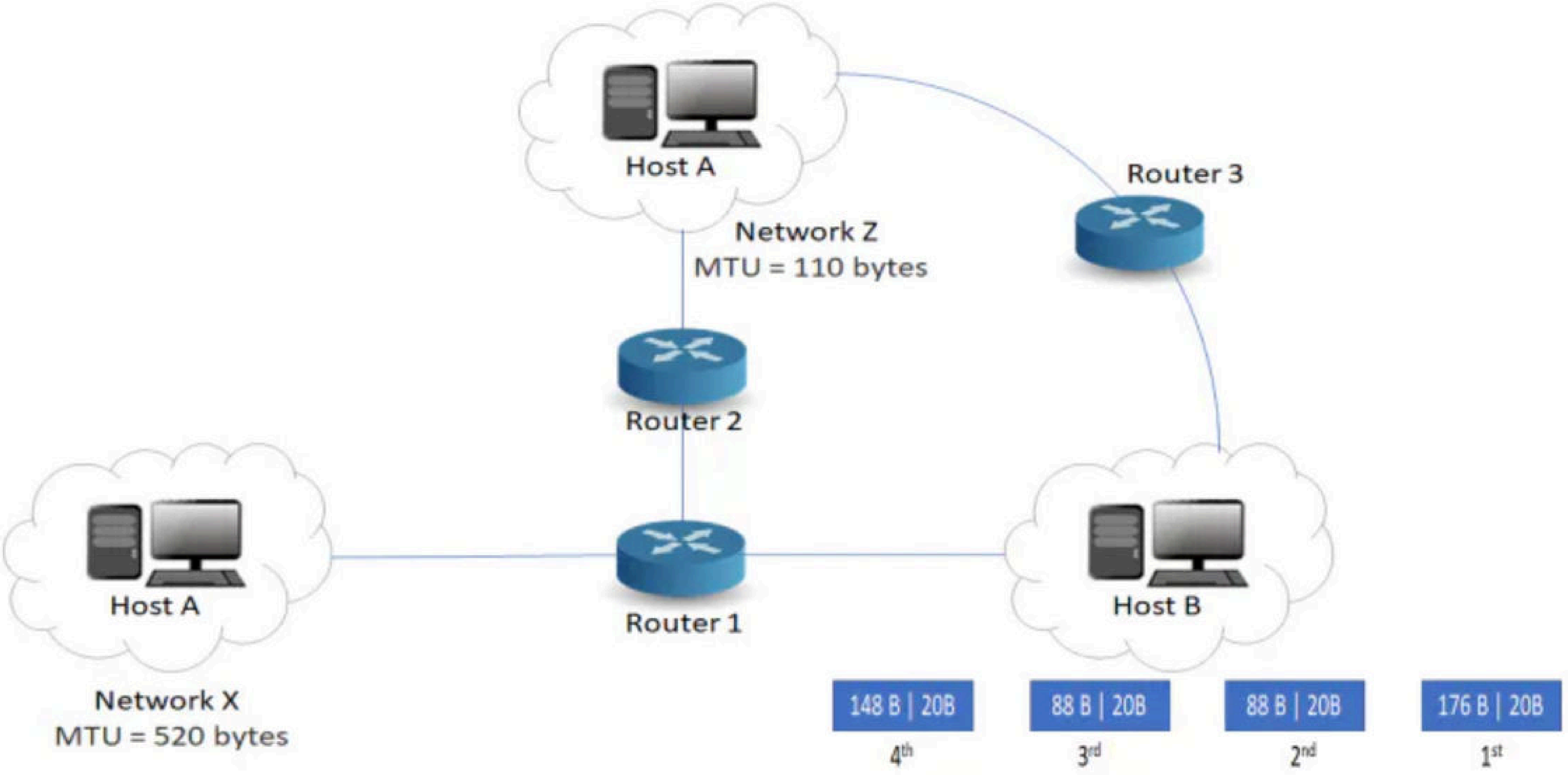
- This fragment is **NOT** the last fragment of the original datagram.
- It is the last fragment of the datagram received by Router-2.
- The datagram received by Router-2 is the second fragment of the original datagram.
- This datagram will serve as the third fragment of the original datagram.
- There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

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



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.