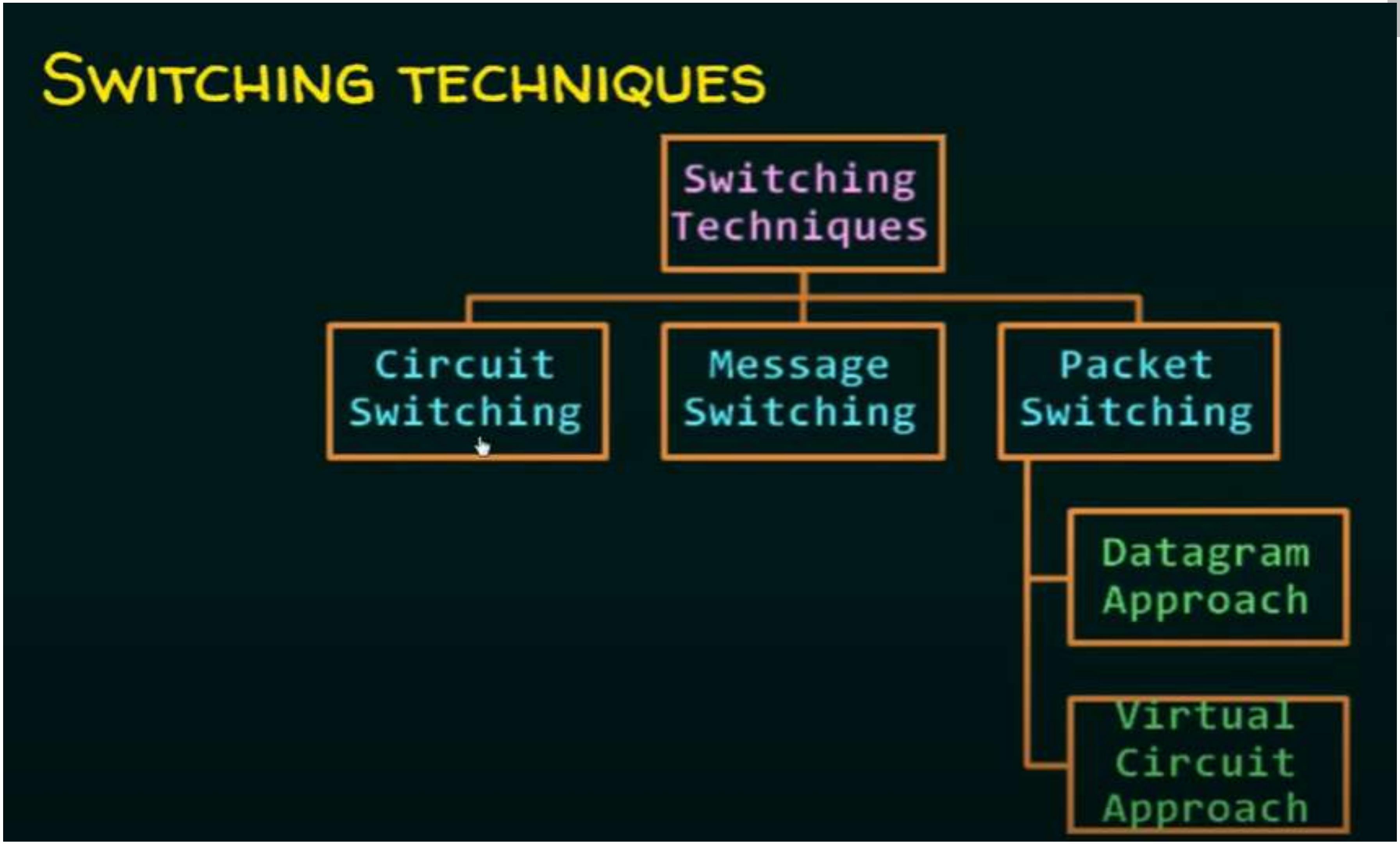
UNIT – III

- **Network Layer:** Switching, Logical addressing - IPV4, IPV6; Address mapping - ARP, RARP, BOOTP and DHCP-Delivery, Forwarding and Unicast Routing protocols
- **Transport Layer:** Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), SCTP Congestion Control; Quality of Service (QoS), QoS improving techniques - Leaky Bucket and Token Bucket algorithms

What is Switching?

- Switching in computer networks refers to the process of directing data packets from a source to a destination across a network.
- It is a fundamental concept that enables efficient communication between devices.
- Switching techniques help optimize network performance, reduce congestion, and ensure reliable data transmission.

Switching Techniques



SWITCHING TECHNIQUES

Scenario

Circuit Switching

resource reservation

Restaurant A

accepts reservation

Packet Switching

no resource reservation

Restaurant B

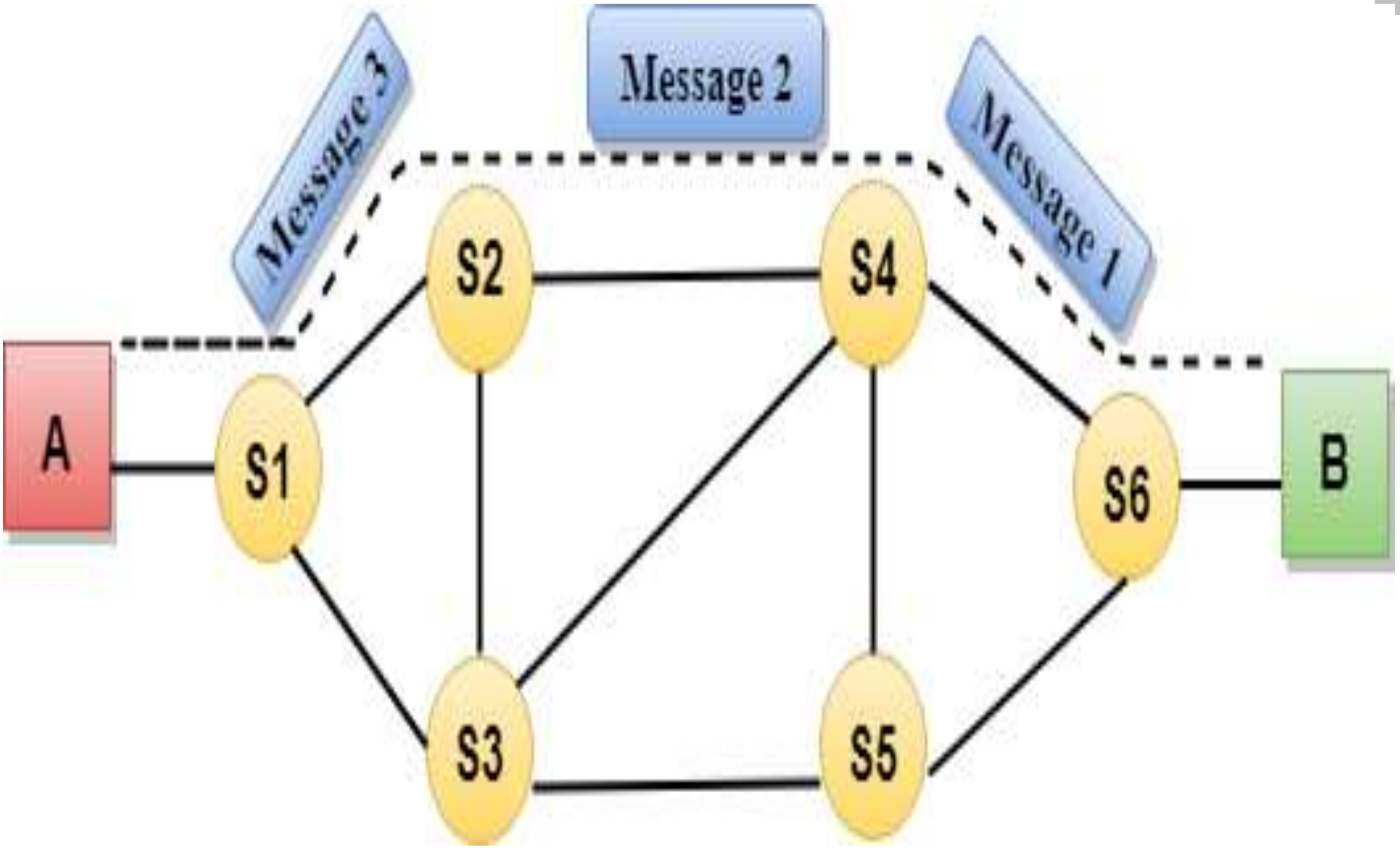
no reservation

1. Circuit Switching

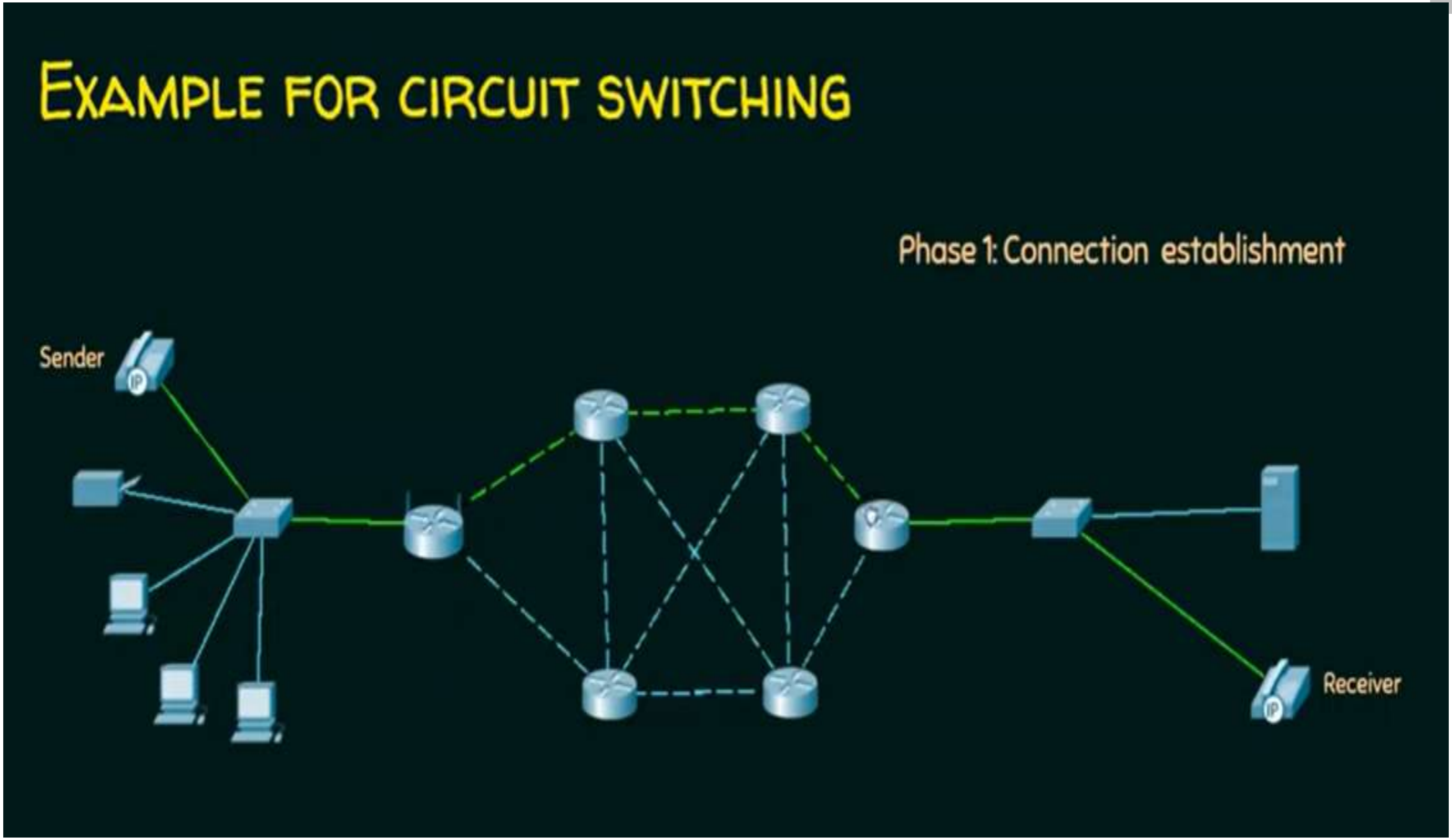
- A dedicated path is established between the sender and the receiver.
- Before data transfer, connection will be established first
- **Example : Telephone Network**

3 Phases in Circuit Switching

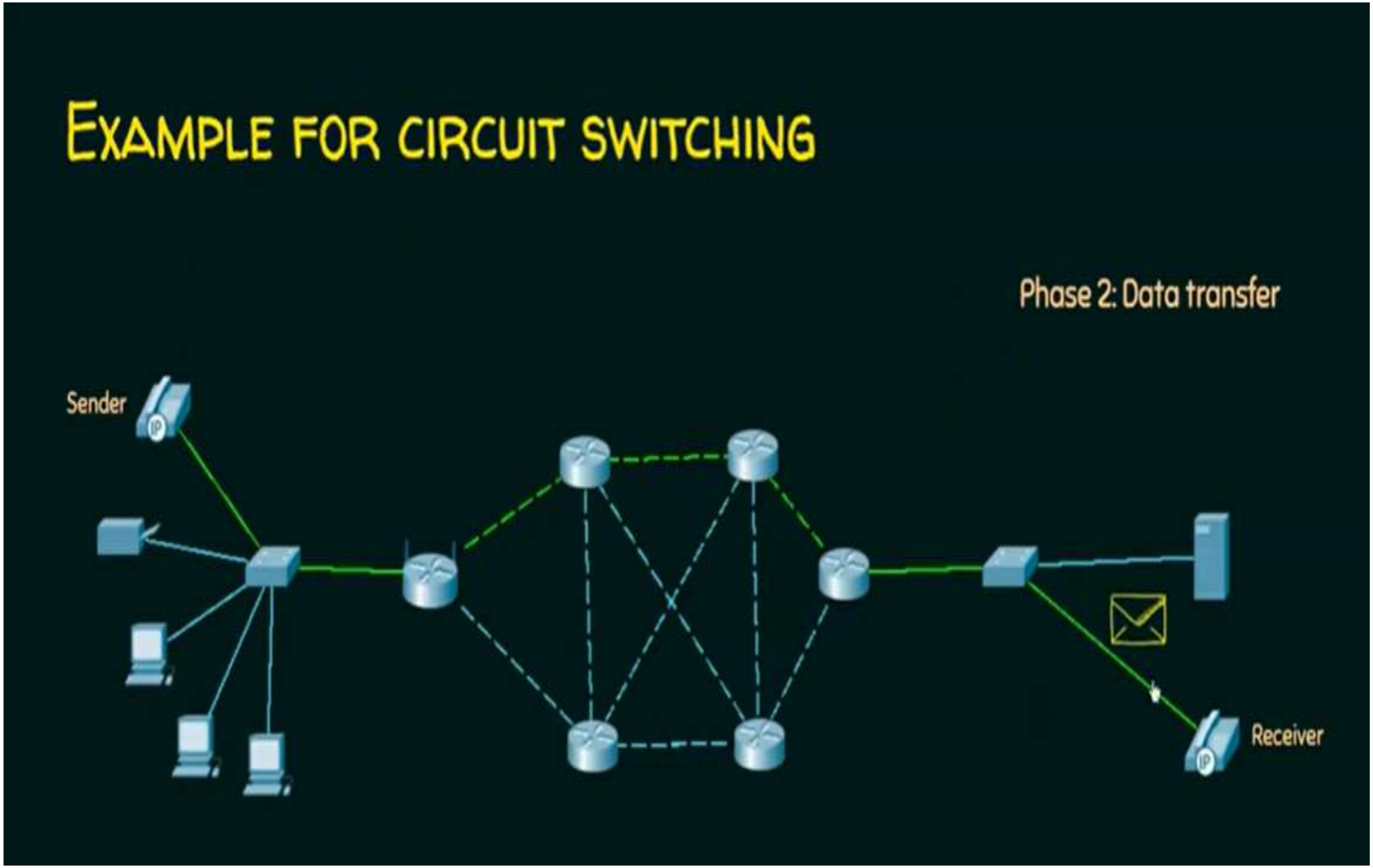
1. Connection establishment
2. Data transfer
3. Connection disconnection


Message 3
Message 2
Message 1
A
S1
S2
S3
S4
S5
S6
B

Connection Establishment


EXAMPLE FOR CIRCUIT SWITCHING
Phase 1: Connection establishment
Sender
IP
IP
Receiver

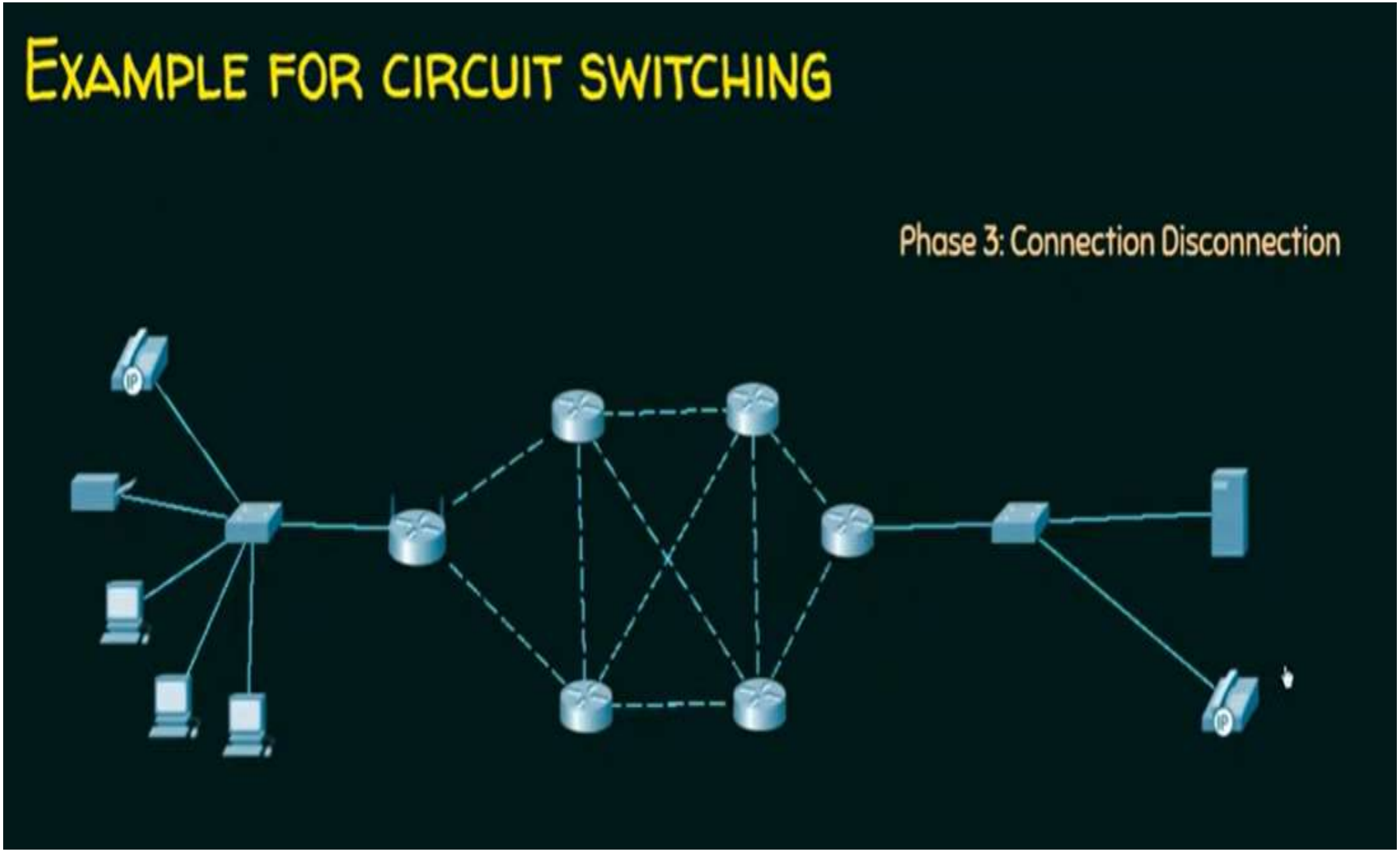
Data transfer



EXAMPLE FOR CIRCUIT SWITCHING

Phase 2: Data transfer

Connection disconnection



EXAMPLE FOR CIRCUIT SWITCHING

Phase 3: Connection Disconnection

Advantages Of Circuit Switching:

- Dedicated Communication channel
- Fixed bandwidth

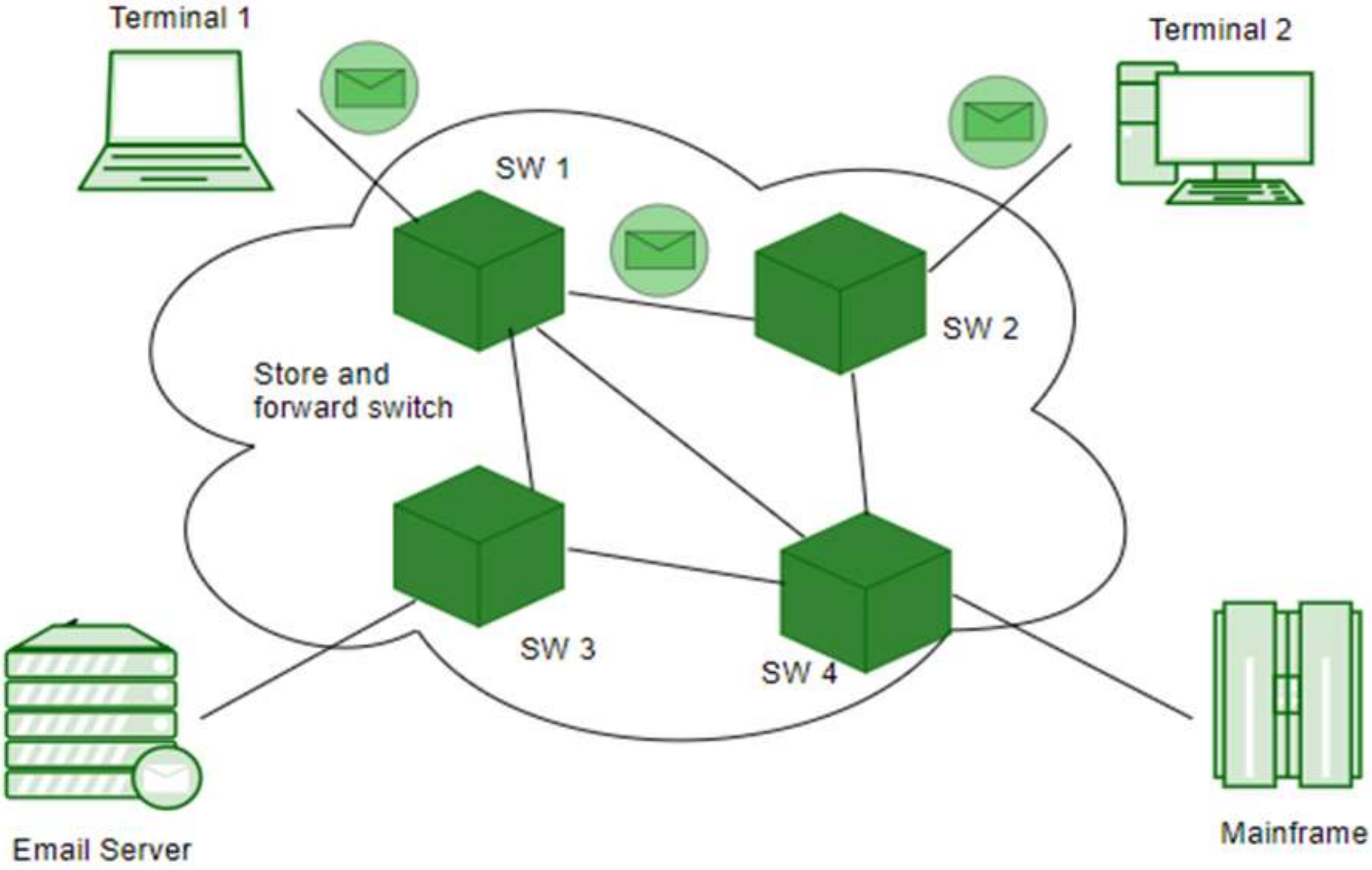
Disadvantages Of Circuit Switching:

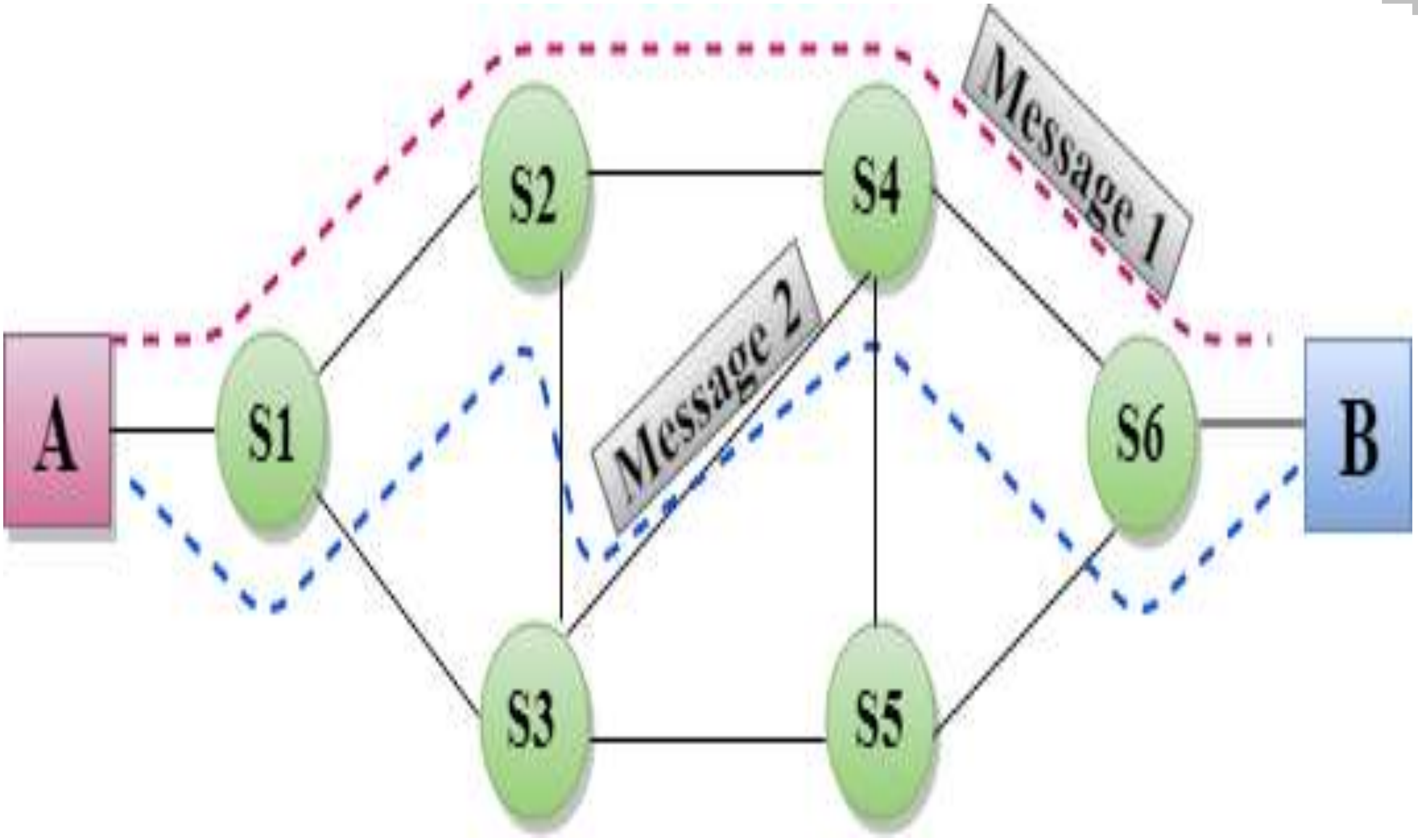
- Once the dedicated path is established, the only delay occurs in the speed of data transmission.
- It takes a long time to establish a connection approx **10 seconds** during which no data can be transmitted.
- It is more expensive than other switching techniques as a dedicated path is required for each connection.
- It is inefficient to use because once the path is established and no data is transferred, then the capacity of the path is wasted.
- In this case, the connection is dedicated therefore no other data can be transferred even if the channel is free.

2. Message Switching

- Entire messages are received, stored, and forwarded to the next node.
- No dedicated path; messages are temporarily held in intermediate nodes.
- Example: Email and SMS networks.

- Each and every node stores the entire message and then forward it to the next node. This type of network is known as store and forward network.
- Message switching treats each message as an independent entity.
- Not suited for streaming media and real time applications


Terminal 1
Terminal 2
SW 1
SW 2
Store and forward switch
SW 3
SW 4
Email Server
Mainframe


Message 1
Message 2
A
S1
S2
S3
S4
S5
S6
B

Advantages Of Message Switching

- Data channels are shared among the communicating devices that improve the efficiency of using available bandwidth.
- Traffic congestion can be reduced because the message is temporarily stored in the nodes.
- Message priority can be used to manage the network.
- Supports the data of unlimited size.

Disadvantages Of Message Switching

- The message switches must be equipped with sufficient storage to enable them to store the messages until the message is forwarded.
- The Long delay can occur due to the storing and forwarding facility provided by the message switching technique.

3. Packet Switching

- The internet is a packet switched network.
- Message is broken into individual chunks called **“Packets”.**
- Each packet is sent individually.
- Each packet will have source and destination IP address along with the sequence number.
- Sequence numbers will help the receiver to

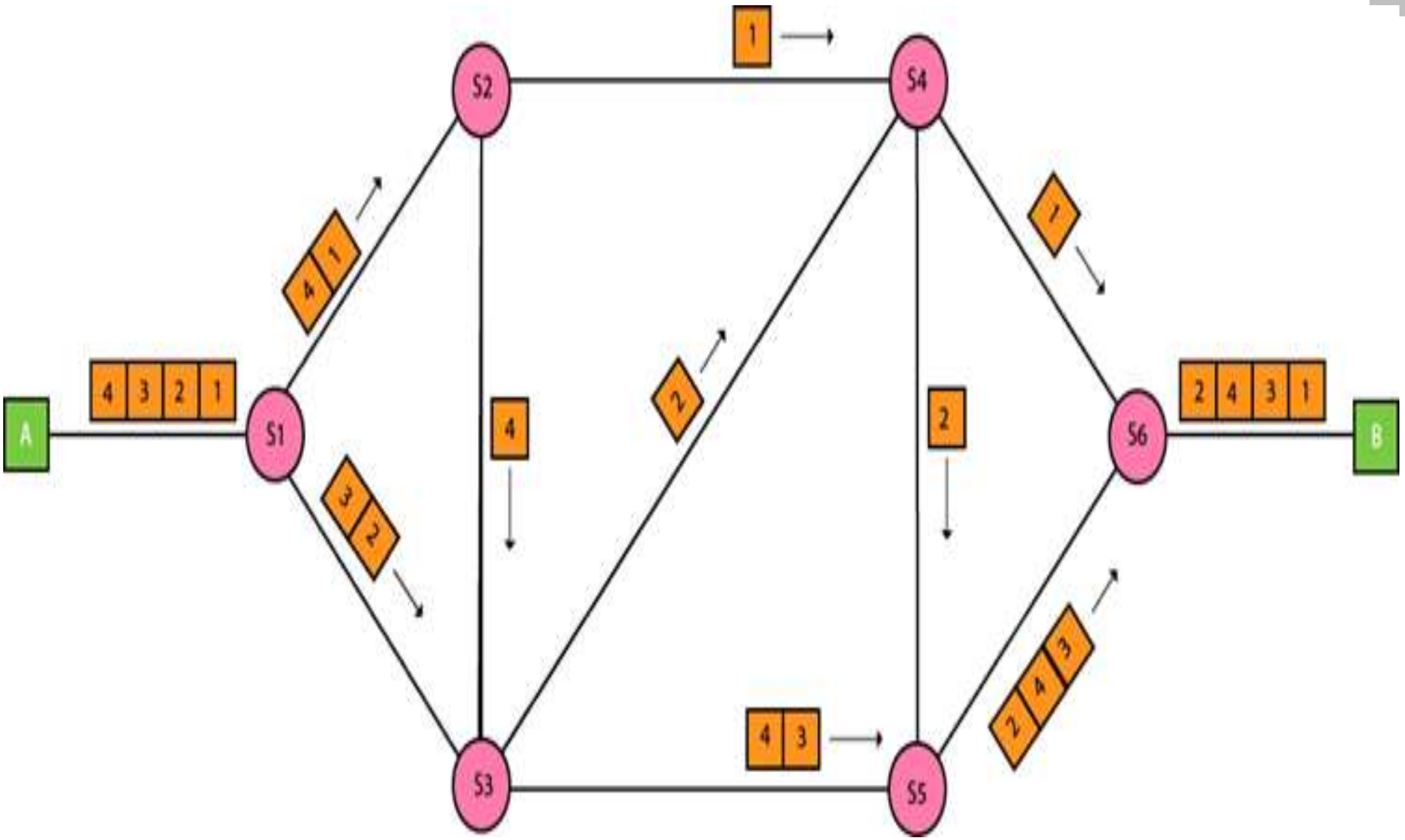
1. Reorder the packets
2. Detect missing packets
3. Send acknowledgement

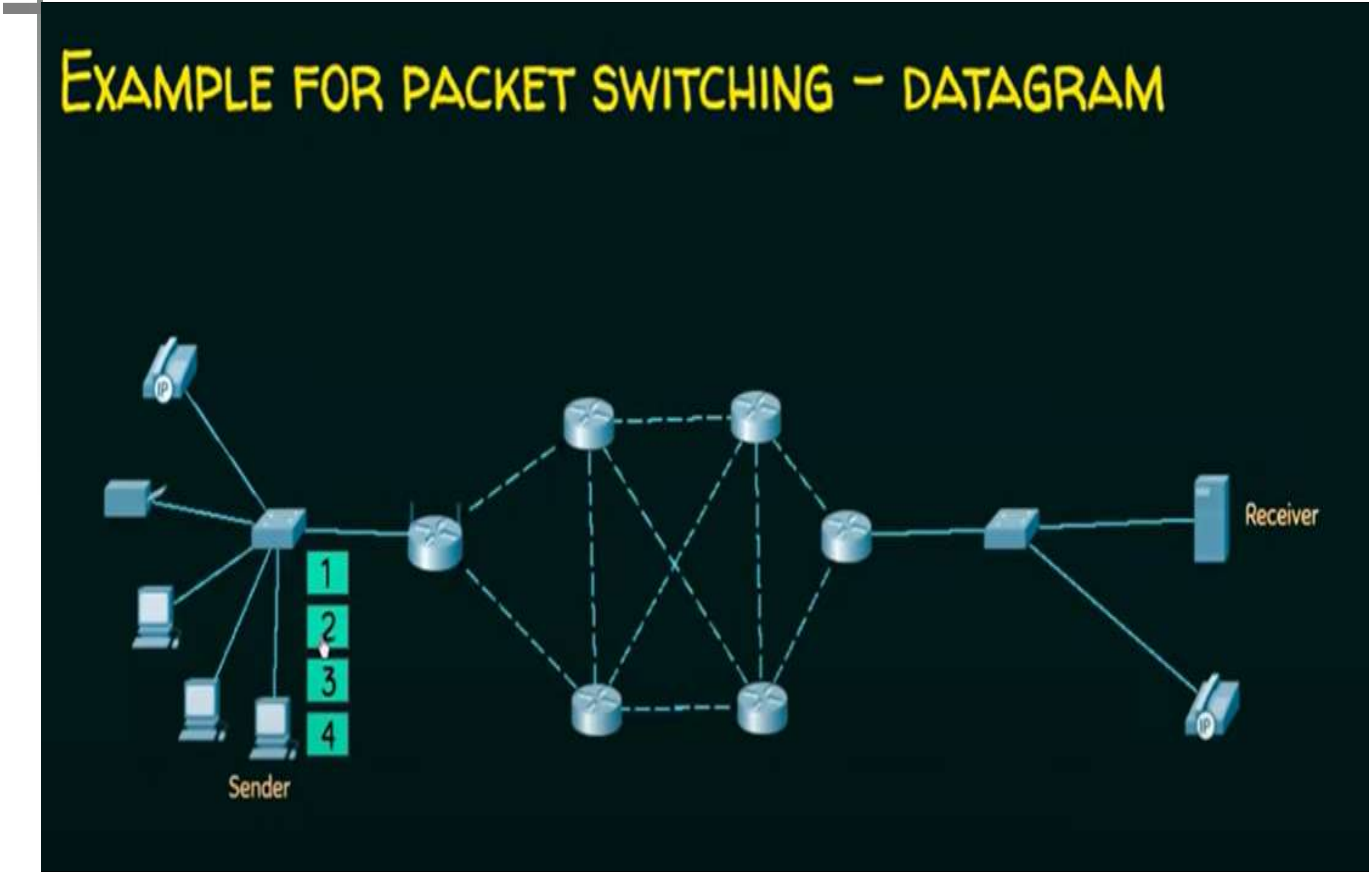
Two approaches to Packet Switching

1. Datagram Approach
2. Virtual Circuit(VC) Approach

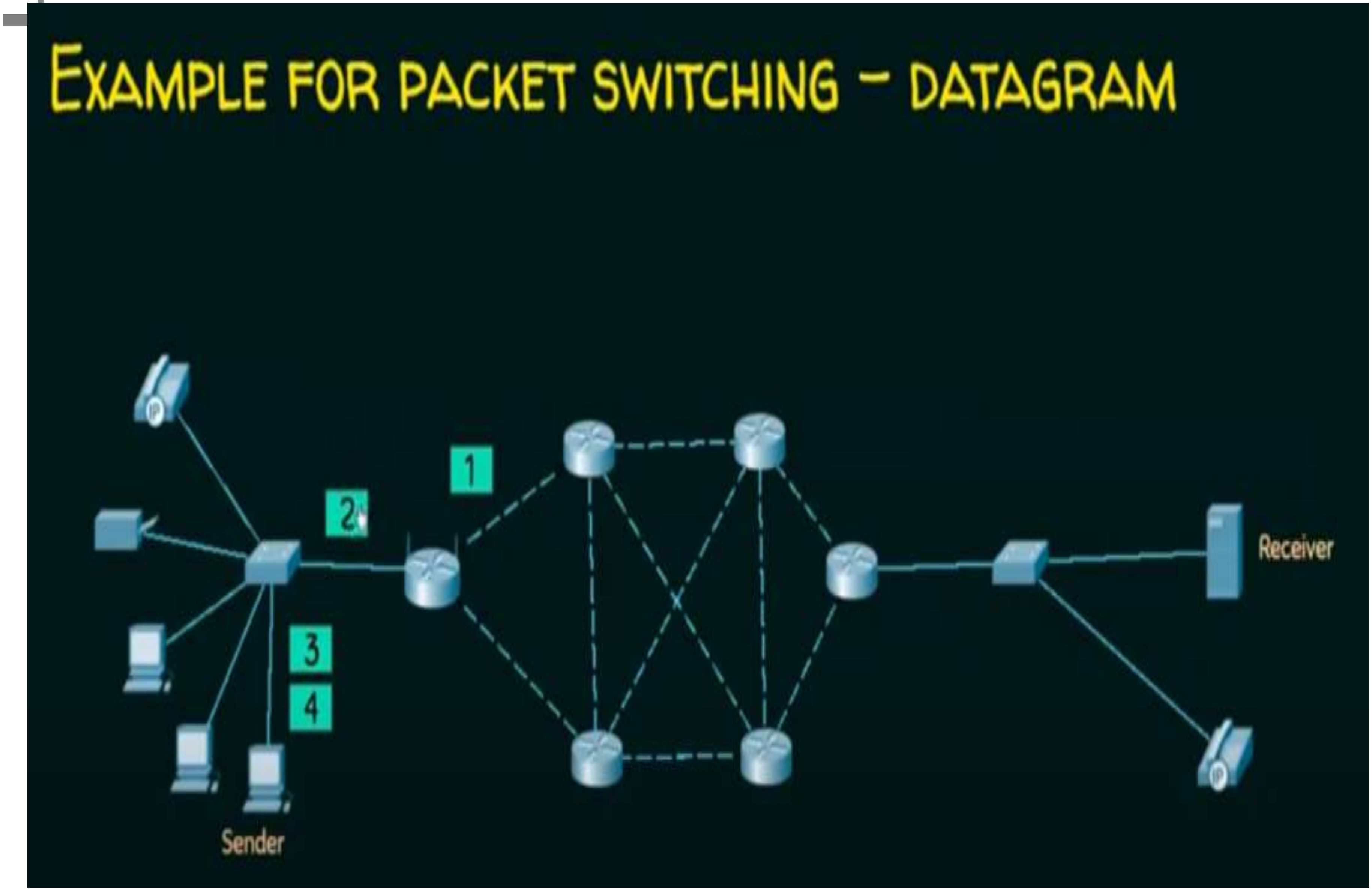
1. Packet Switching – Datagram Approach

- Datagram packet switching is also known as ***Connection-less switching***
- Each independent entity is called as **Datagram**
- Datagram contains destination information and the intermediary nodes/devices use this information to forward datagrams to the right destination
- No Fixed path
- Intermediate nodes take the routing decisions to forward the packets

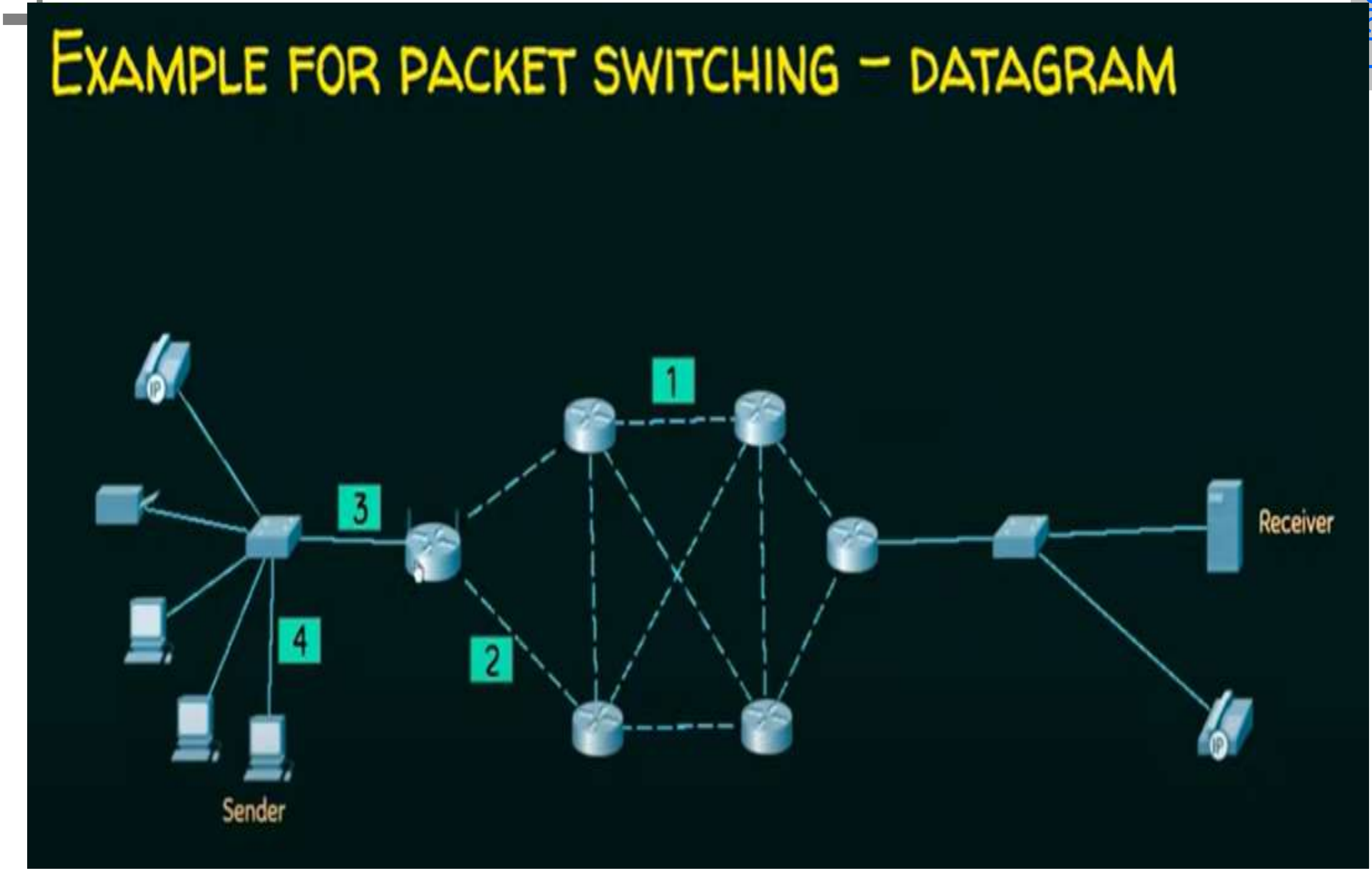

1
S2
S4
4 1
1
A
4 3 2 1
S1
4
2
2
2 4 3 1
S6
B
3 2
S3
4 3
S5
2 4 3



EXAMPLE FOR PACKET SWITCHING – DATAGRAM

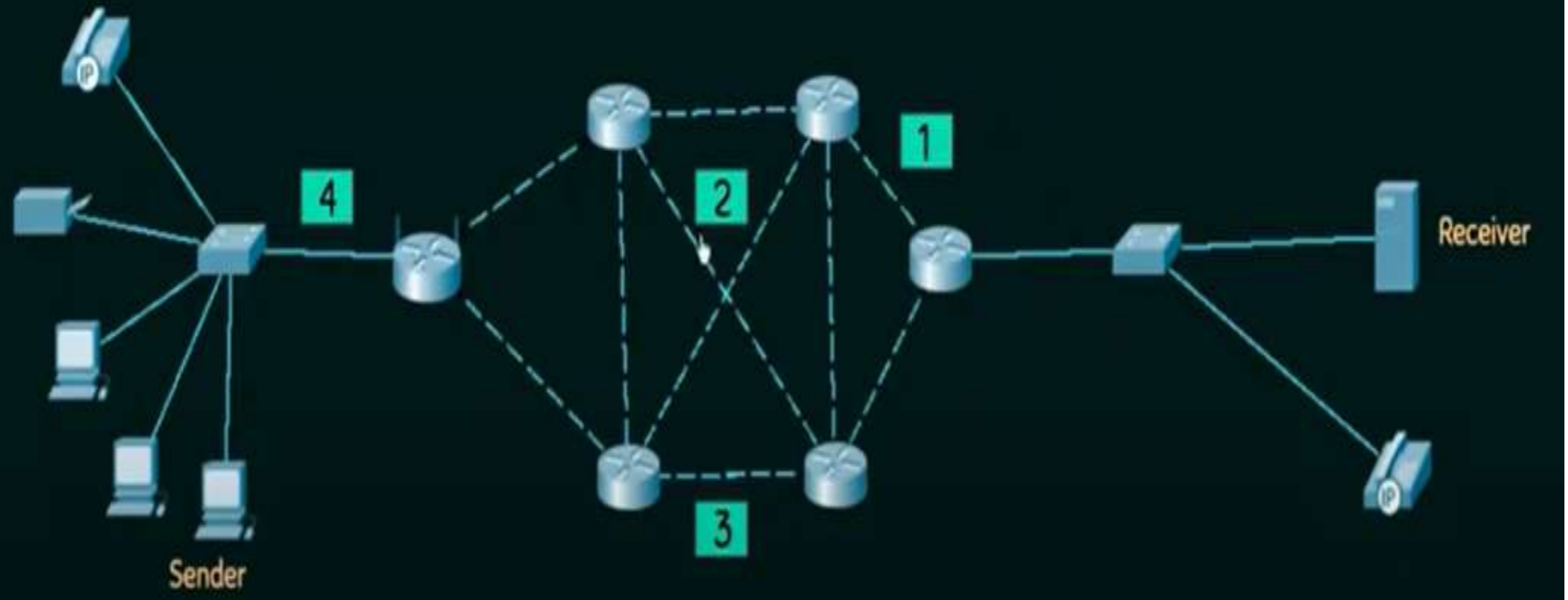


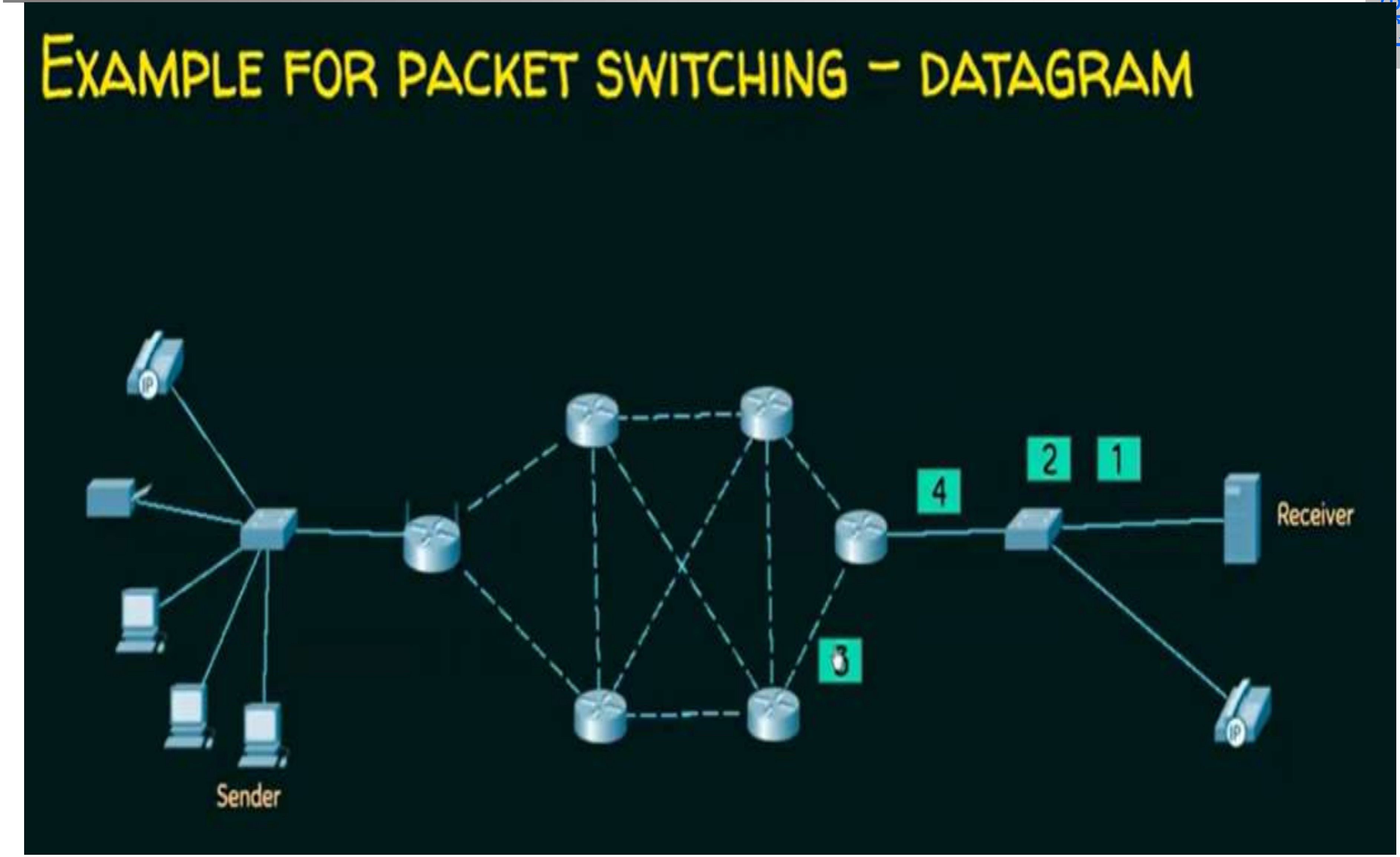
EXAMPLE FOR PACKET SWITCHING – DATAGRAM



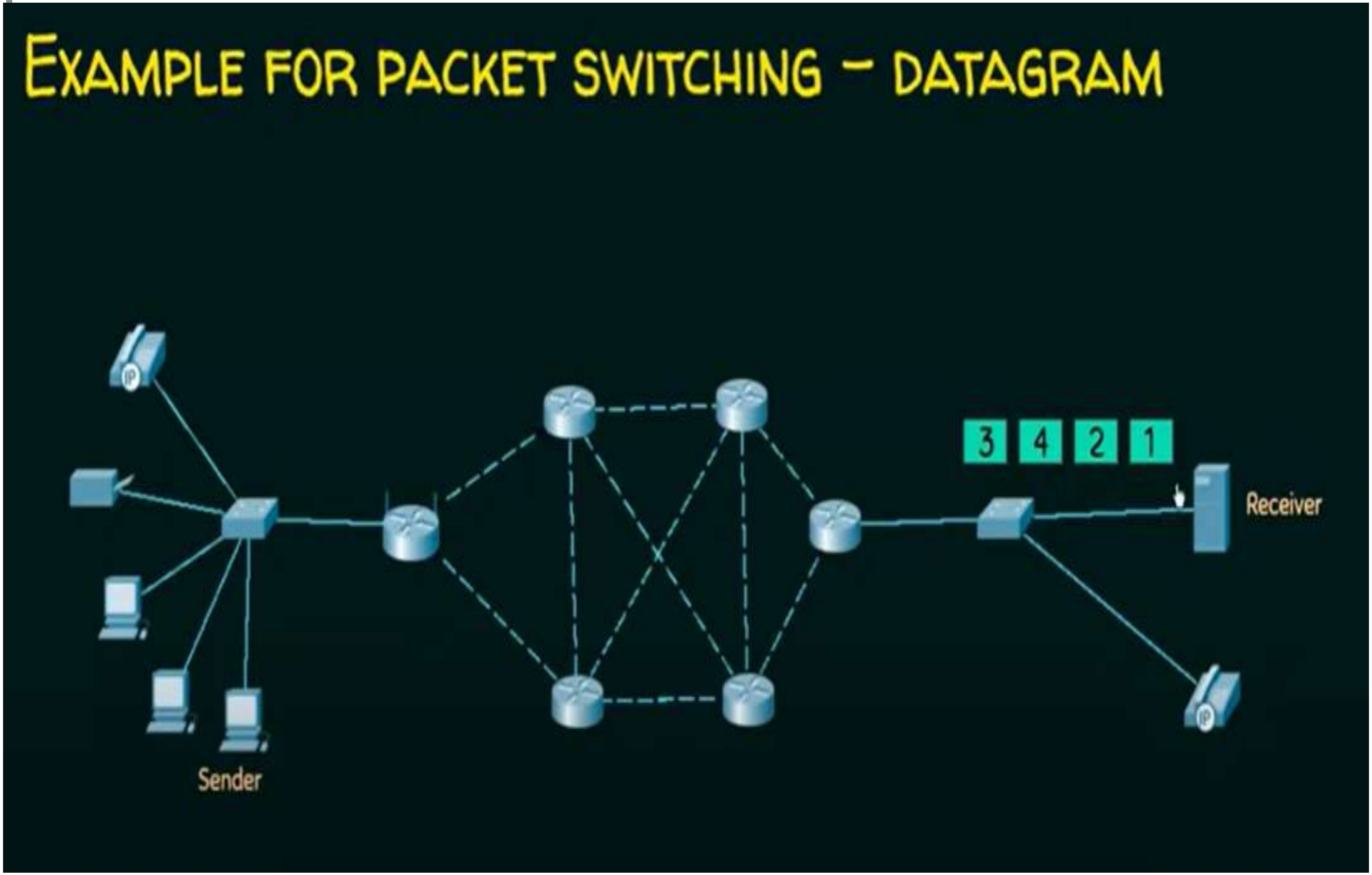
EXAMPLE FOR PACKET SWITCHING – DATAGRAM

EXAMPLE FOR PACKET SWITCHING – DATAGRAM

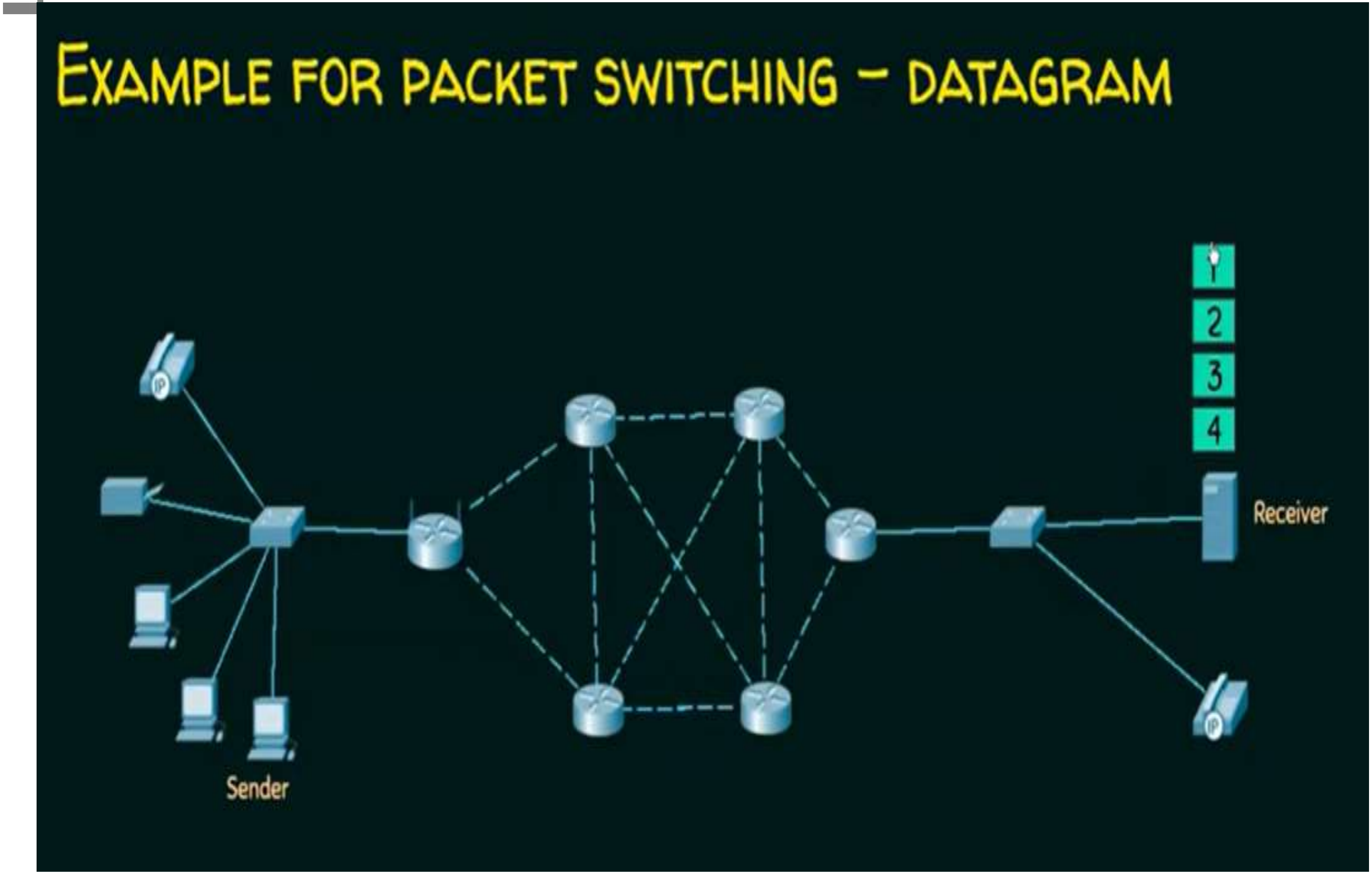




EXAMPLE FOR PACKET SWITCHING – DATAGRAM



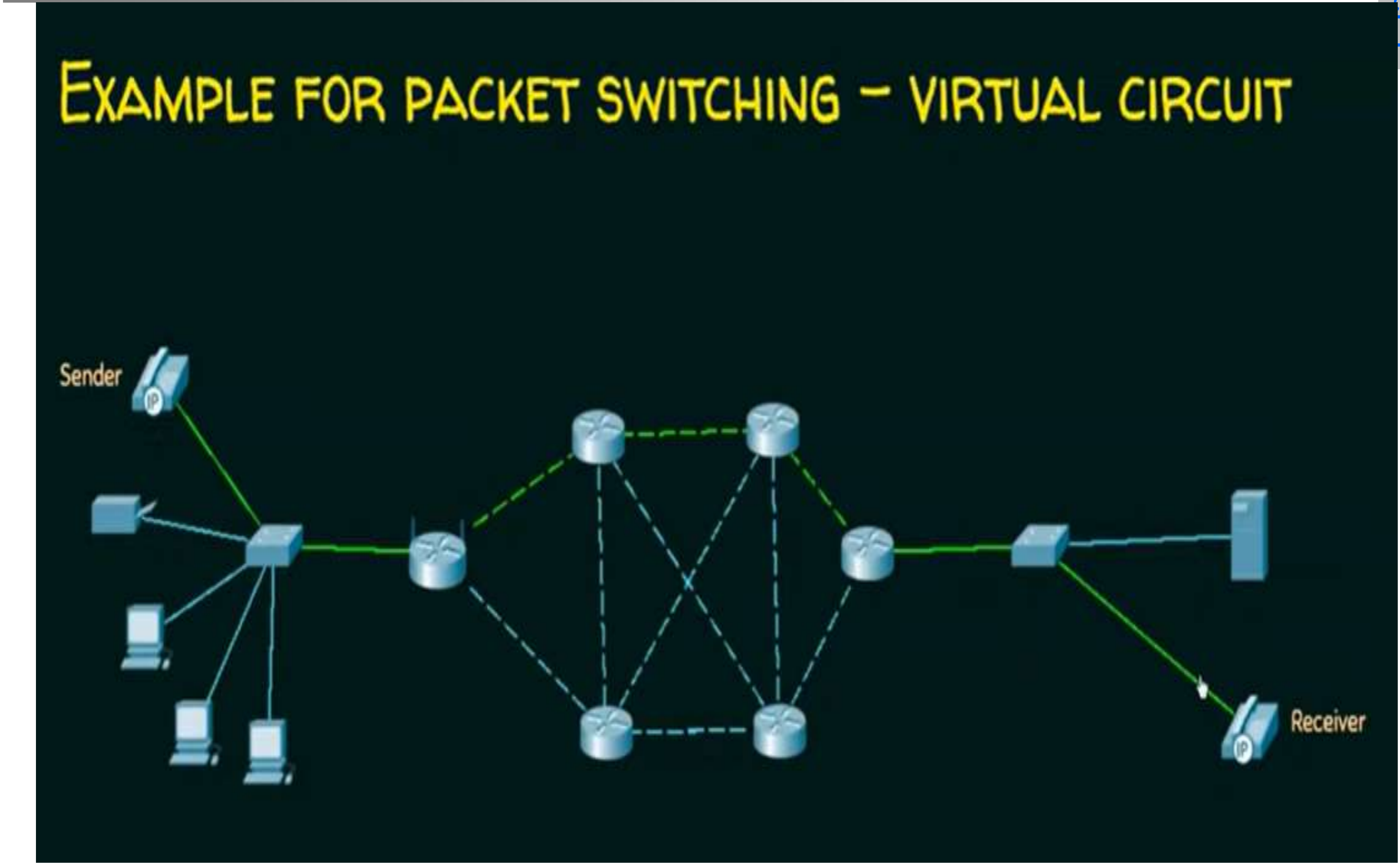
EXAMPLE FOR PACKET SWITCHING – DATAGRAM



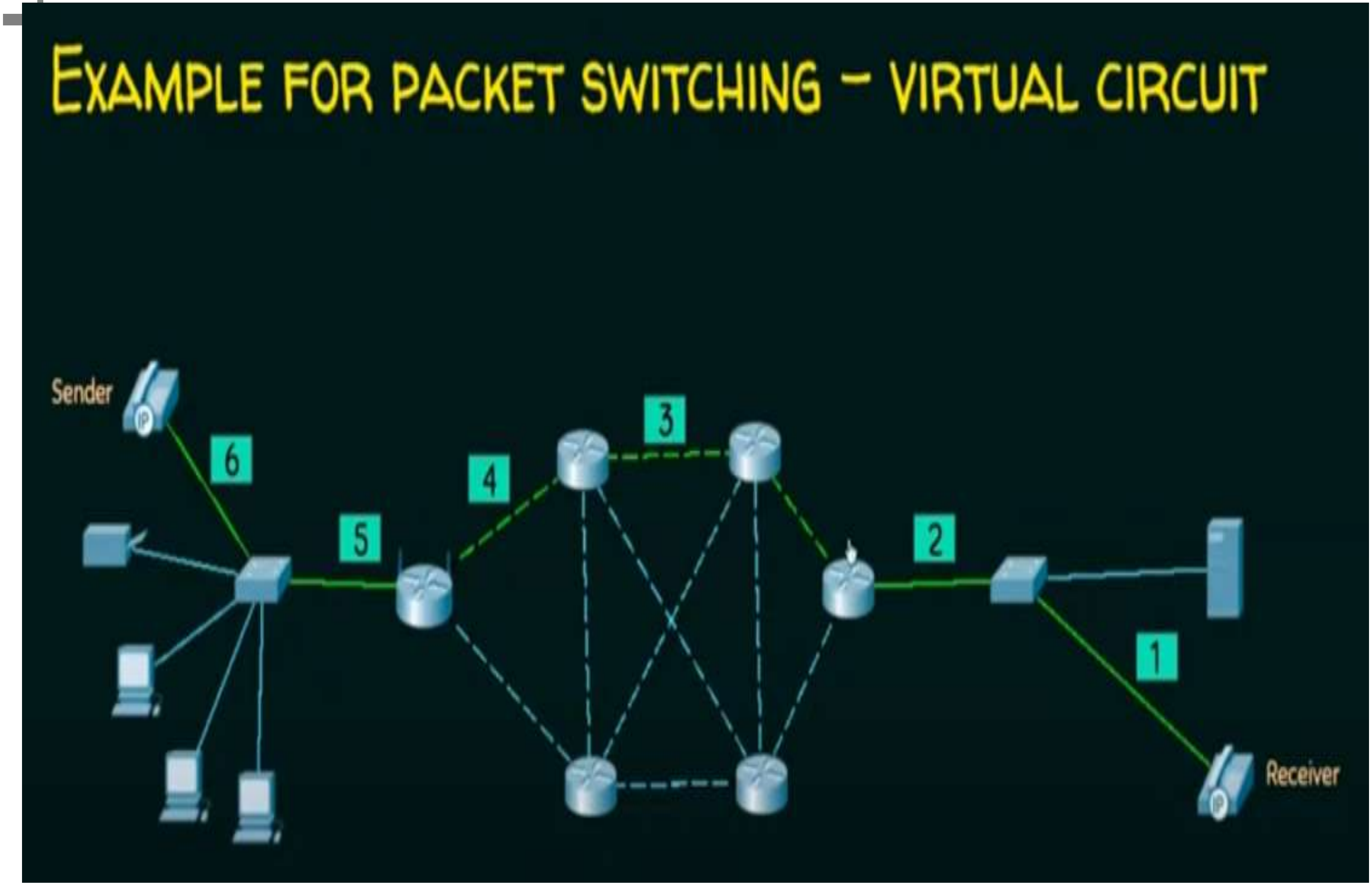
EXAMPLE FOR PACKET SWITCHING – DATAGRAM

2. Packet Switching – Virtual Circuit Approach

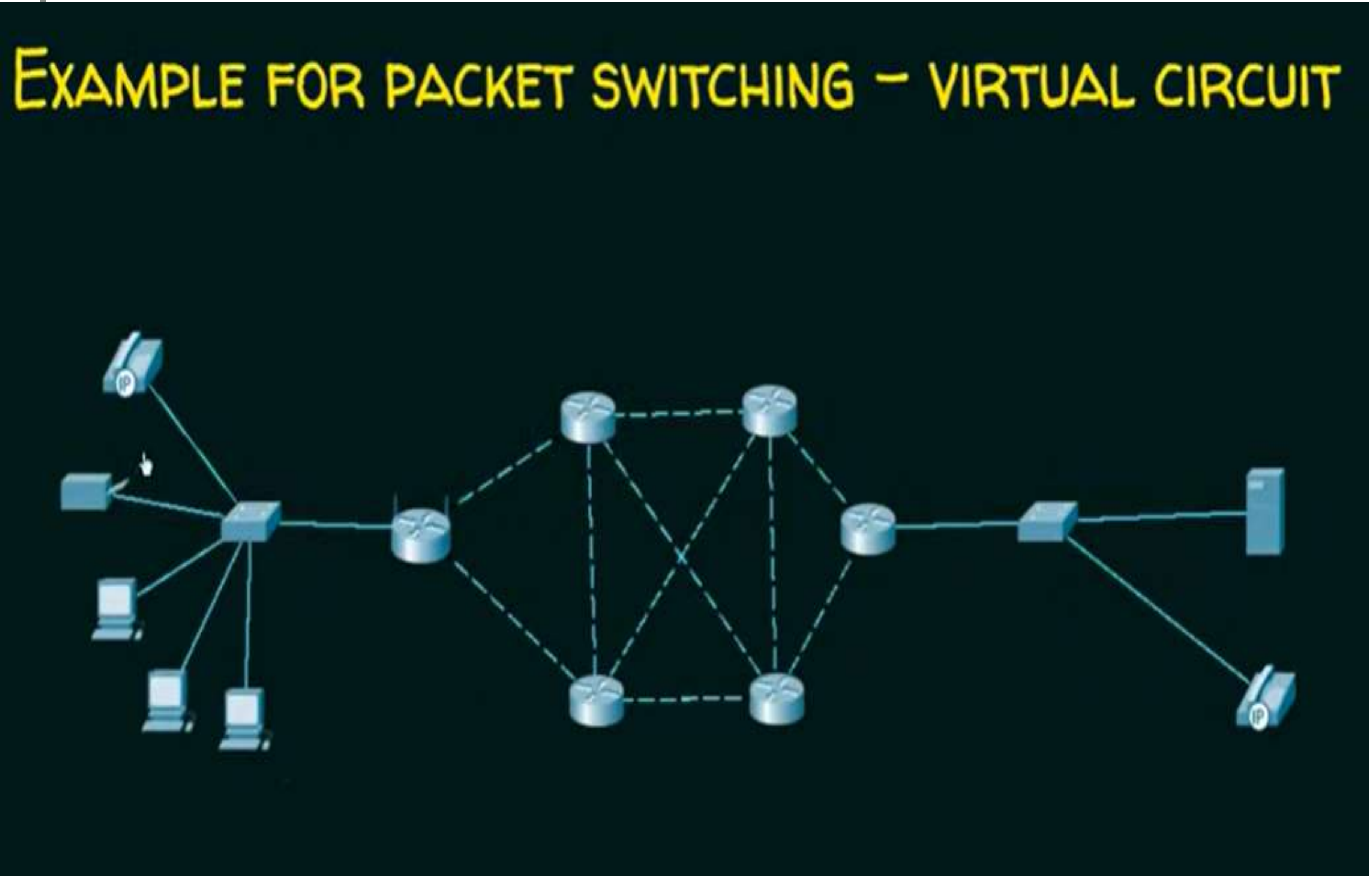
- Also known as Connection – oriented switching
- A pre-planned route is established before the messages are sent
- **Call request and Call accept** packets are used to establish the connection between the sender and the receiver.
- The path is fixed for the duration of a logical connection



EXAMPLE FOR PACKET SWITCHING – VIRTUAL CIRCUIT

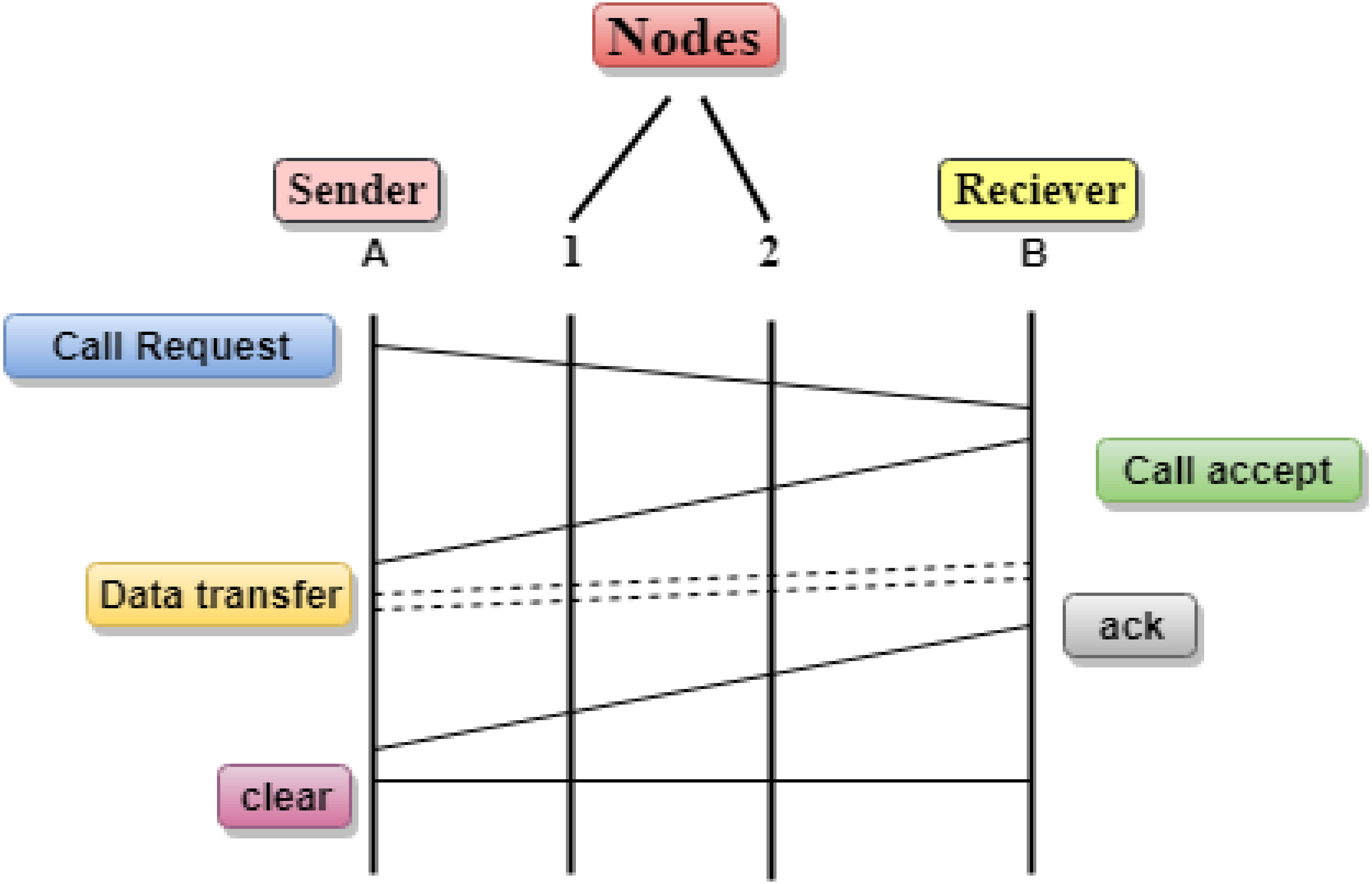


EXAMPLE FOR PACKET SWITCHING – VIRTUAL CIRCUIT

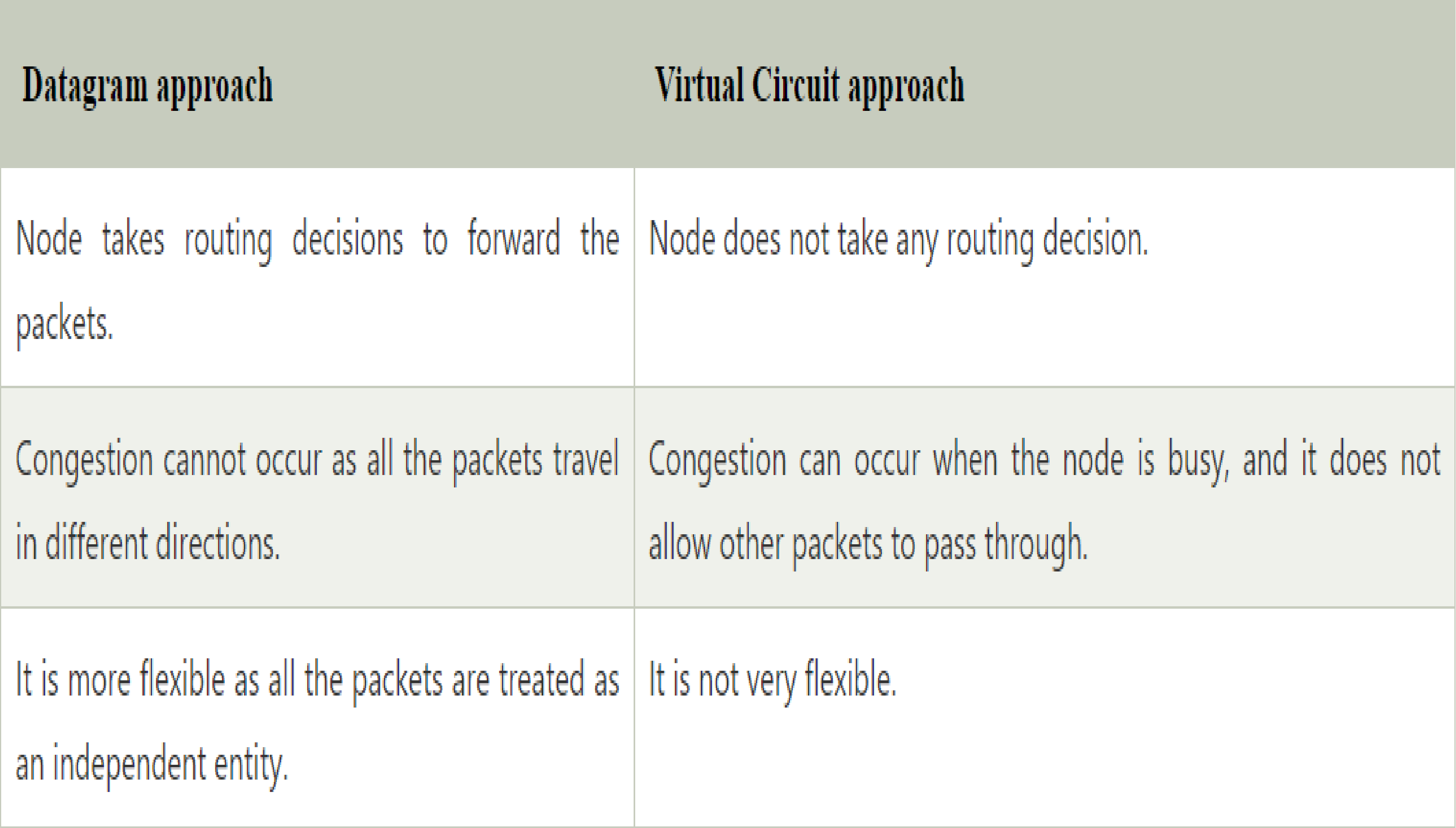


EXAMPLE FOR PACKET SWITCHING – VIRTUAL CIRCUIT

virtual circuit switching



Datagram approach Vs Virtual Circuit approach



Datagram approach	Virtual Circuit approach
Node takes routing decisions to forward the packets.	Node does not take any routing decision.
Congestion cannot occur as all the packets travel in different directions.	Congestion can occur when the node is busy, and it does not allow other packets to pass through.
It is more flexible as all the packets are treated as an independent entity.	It is not very flexible.

Advantages Of Packet Switching:

- **Cost-effective:** In packet switching technique, switching devices do not require massive secondary storage to store the packets, so cost is minimized to some extent.
- **Reliable**: If any node is busy, then the packets can be rerouted.
- **Efficient**: It does not require any established path prior to the transmission, and many users can use the same communication channel simultaneously, hence makes use of available bandwidth very efficiently.

Disadvantages Of Packet Switching:

- Not suitable for applications that require low delay and high-quality services.
- The protocols used in a packet switching technique are very complex and requires high implementation cost.
- Retransmission of lost packets.

QUESTION #1

- Under what conditions would circuit switching be a better network design than packet switching?

SOLUTION

Circuit switching offers

- **Guaranteed capacity**
- **No reordering of packets**
- **Bounded delay**
- **No lost packets**
- Various streaming applications (VOIP, video streaming, real-time monitoring, ...) would benefit from these features.

Question #2

- Circuit switching and packet switching are two different ways of sharing links in a communication network. **Indicate True or False for each choice.**

a) Switches in a circuit-switched network process connection establishment and tear-down messages, whereas switches in a packet-switched network do not.

b) Under some circumstances, a circuit-switched network may prevent some senders from starting new conversations.

c) Once a connection is correctly established, a switch in a circuit-switched network can forward data correctly without requiring data frames to include a destination address.

Solution

a) True

b) True

c) True

Problem 1:

- Consider a network where a **circuit-switched** connection is established between two users. The link has a data rate of **1 Gbps (10^9 bps)**, and each circuit-switched connection requires a fixed **bandwidth of 50 Mbps (50×10^6 bps)**.

1. How many simultaneous connections can be supported on this link?
2. If each connection lasts **5 minutes**, how many users can be served in **1 hour** assuming 100% utilization?

Solution

1. Number of Simultaneous Connections

- The total bandwidth available = **1 Gbps**
  Each connection requires = **50 Mbps**
- Number of connections=Total Bandwidth/Bandwidth per connection
- Total Bandwidth =10^9 bps / $50×10^6$ bps= 20 connections
- So, the link can support **20 simultaneous connections**.

2. Number of Users Served in 1 Hour

- Each connection lasts **5 minutes**, so in **1 hour (60 minutes)**, the number of **time slots** available per connection is:
- 60/5=12 sessions per hour per circuit
- Since there are **20 circuits**, the total number of users served per hour is:
- **20×12=240 users**

Problem 2:

- A **10 MB (Megabyte)** file needs to be sent over a **packet-switched network**. The network link has a transmission rate of **10 Mbps (Megabits per second)**, and the packet size is **1 KB (Kilobyte)**. The propagation delay is **10 ms (milliseconds)**, and each packet has a processing delay of **2 ms** at the router. Assume no queuing delay.
- **Find:**

1. **Number of packets required to transmit the file**
2. **Time taken to transmit one packet**
3. **Total transmission time for the entire file**

Solution

1. Number of Packets Required

- The file size is **10 MB** (Megabytes), and each packet is **1 KB** (Kilobyte).
  Converting to the same unit:
- 10MB=10×1024KB=10240KB
- each packet is **1 KB**, the number of packets required:
- 10240 KB / 1 KB per packet=10240 packets

2. Time Taken to Transmit One Packet

- Each packet size = **1 KB** = **8 × 1024 bits** = **8192 bits**
  Transmission rate = **10 Mbps = 10 × 10^6 bps**
- Transmission time per packet=Packet size (bits)/Transmission rate
- 8192/10^7=0.0008192 seconds=0.8192 ms
- Each packet also experiences:
- **Propagation delay** = **10 ms**
- **Processing delay at router** = **2 ms**
- Total delay per packet=0.8192+10+2=**12.8192 ms**

3. Total Transmission Time for Entire File

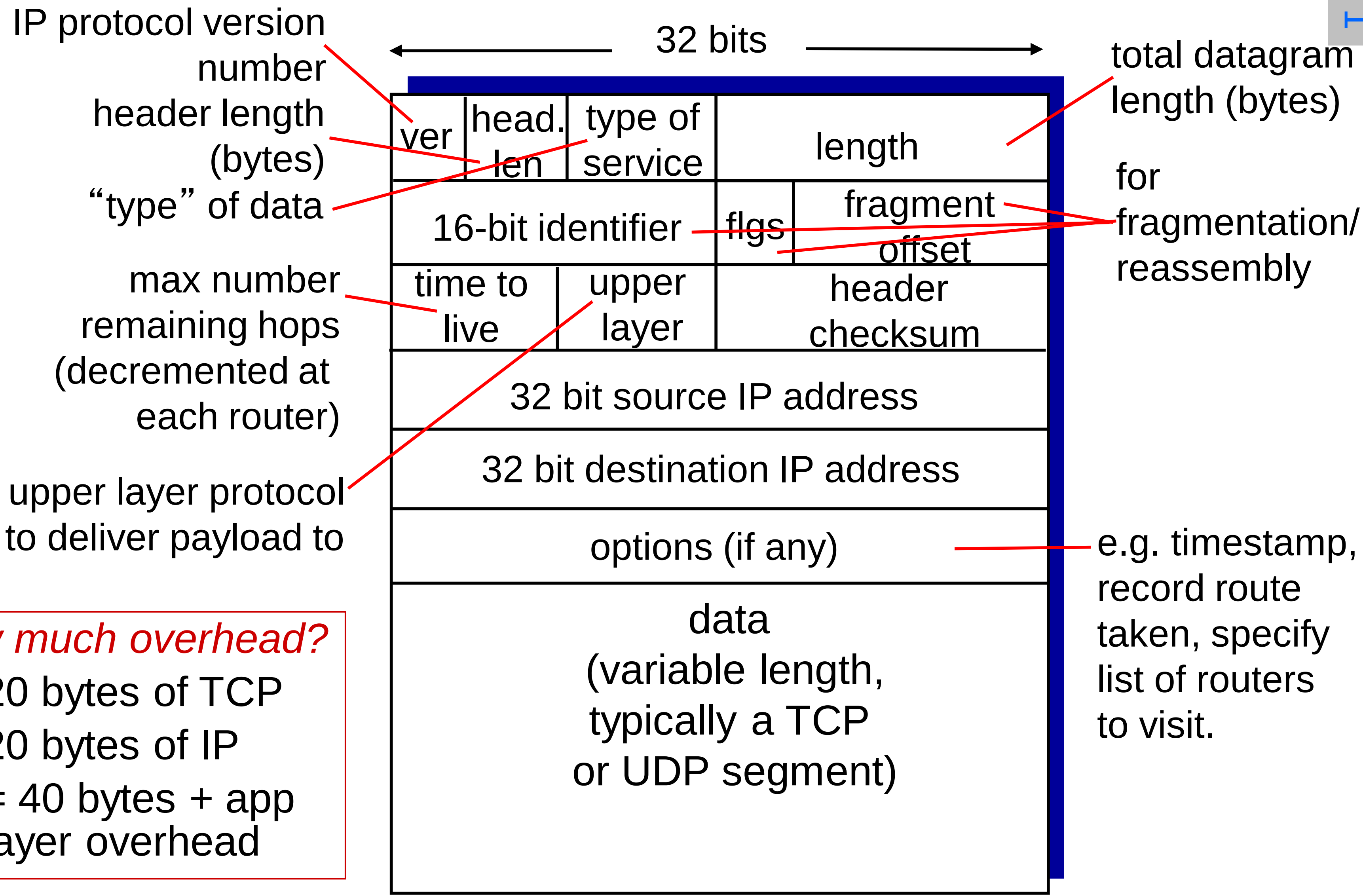
- Total packets = **10240**
  Total time = **(Total packets) × (Delay per packet)**
- 10240×12.8192 ms=131248.3 ms=**131.25 seconds**

Logical addressing - IPV4, IPV6

Logical addressing

- Logical addressing is a fundamental concept in networking that enables devices to be uniquely identified across different networks.
- The two main versions of Internet Protocol (IP) used for logical addressing are **IPv4** and **IPv6**.

IP datagram format



IP protocol version number

header length (bytes)

"type" of data

max number remaining hops (decremented at each router)

upper layer protocol to deliver payload to

32 bits

ver

head. len

type of service

length

16-bit identifier

flgs

fragment offset

time to live

upper layer

header checksum

32 bit source IP address

32 bit destination IP address

options (if any)

data (variable length, typically a TCP or UDP segment)

total datagram length (bytes)

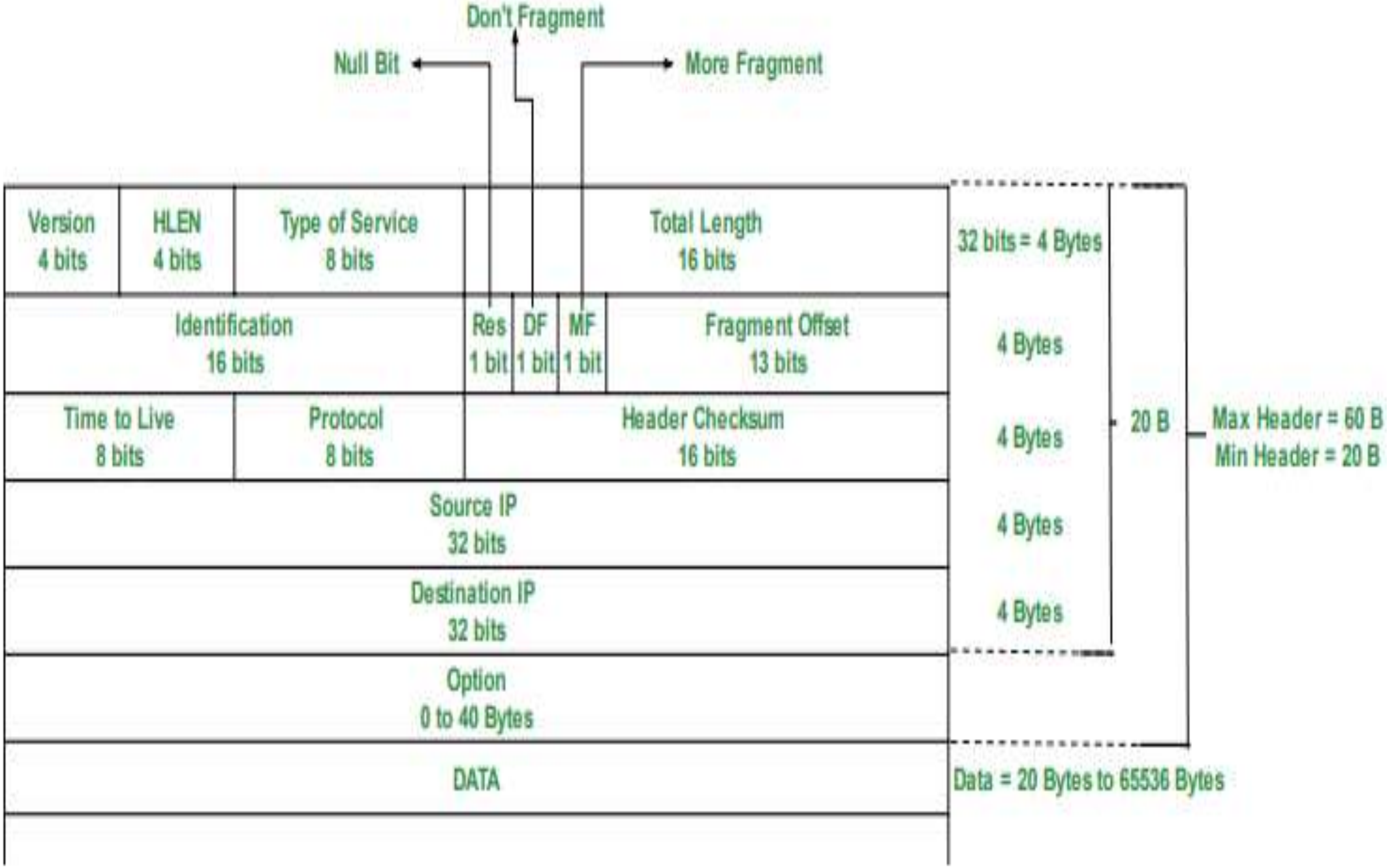
for fragmentation/ reassembly

e.g. timestamp, record route taken, specify list of routers to visit.

how much overhead?

- 20 bytes of TCP
- 20 bytes of IP
- = 40 bytes + app layer overhead

Cha



Don't Fragment

Null Bit

More Fragment

Version 4 bits | HLEN 4 bits | Type of Service 8 bits | Total Length 16 bits

Identification 16 bits | Res 1 bit | DF 1 bit | MF 1 bit | Fragment Offset 13 bits

Time to Live 8 bits | Protocol 8 bits | Header Checksum 16 bits

Source IP 32 bits

Destination IP 32 bits

Option 0 to 40 Bytes

DATA

32 bits = 4 Bytes

4 Bytes

4 Bytes

4 Bytes

4 Bytes

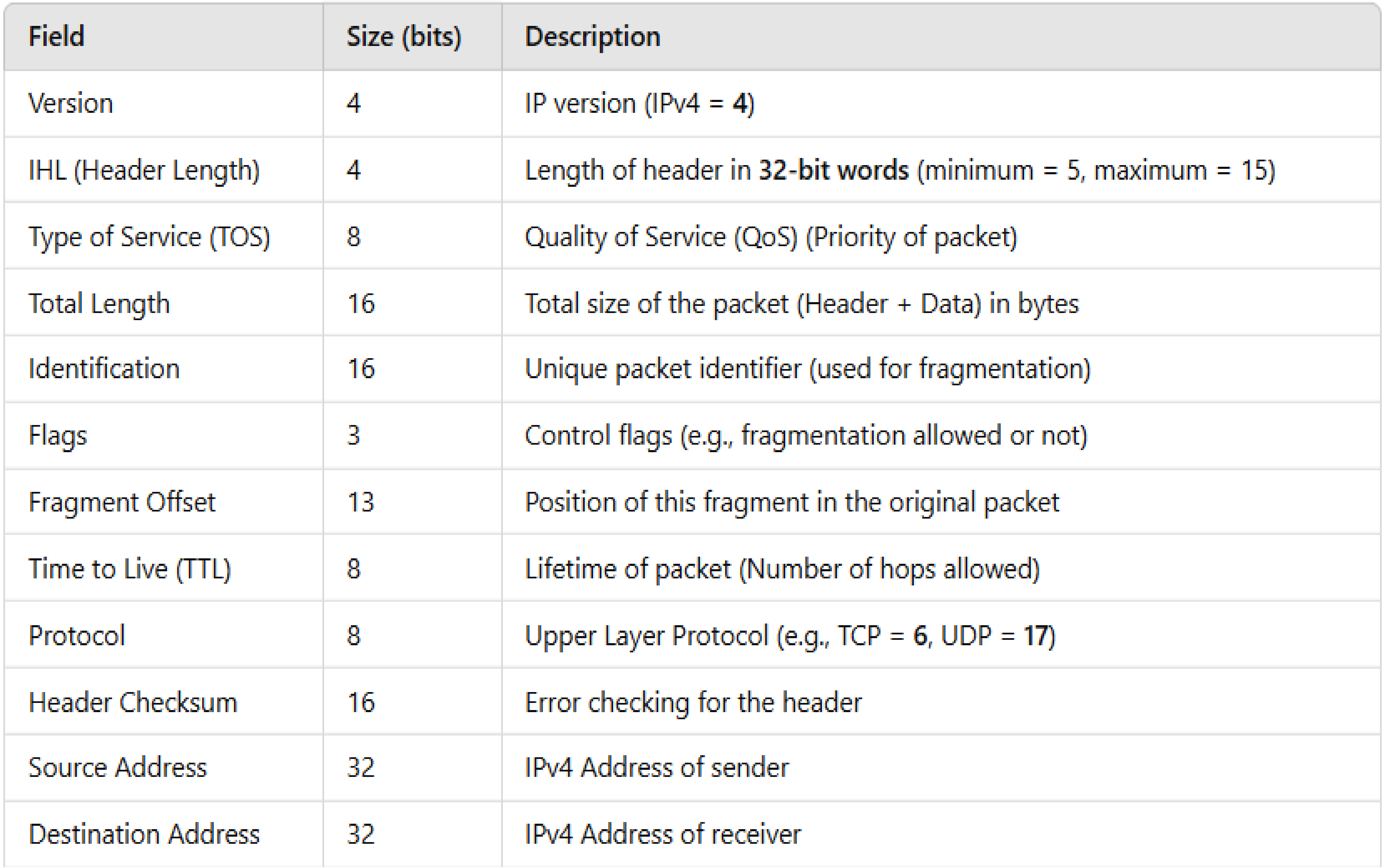
20 B

Max Header = 60 B

Min Header = 20 B

Data = 20 Bytes to 65536 Bytes

IPv4 Packet Header Format



Field	Size (bits)	Description
Version	4	IP version (IPv4 = **4**)
IHL (Header Length)	4	Length of header in **32-bit words** (minimum = 5, maximum = 15)
Type of Service (TOS)	8	Quality of Service (QoS) (Priority of packet)
Total Length	16	Total size of the packet (Header + Data) in bytes
Identification	16	Unique packet identifier (used for fragmentation)
Flags	3	Control flags (e.g., fragmentation allowed or not)
Fragment Offset	13	Position of this fragment in the original packet
Time to Live (TTL)	8	Lifetime of packet (Number of hops allowed)
Protocol	8	Upper Layer Protocol (e.g., TCP = **6**, UDP = **17**)
Header Checksum	16	Error checking for the header
Source Address	32	IPv4 Address of sender
Destination Address	32	IPv4 Address of receiver

Key Features of IPv4

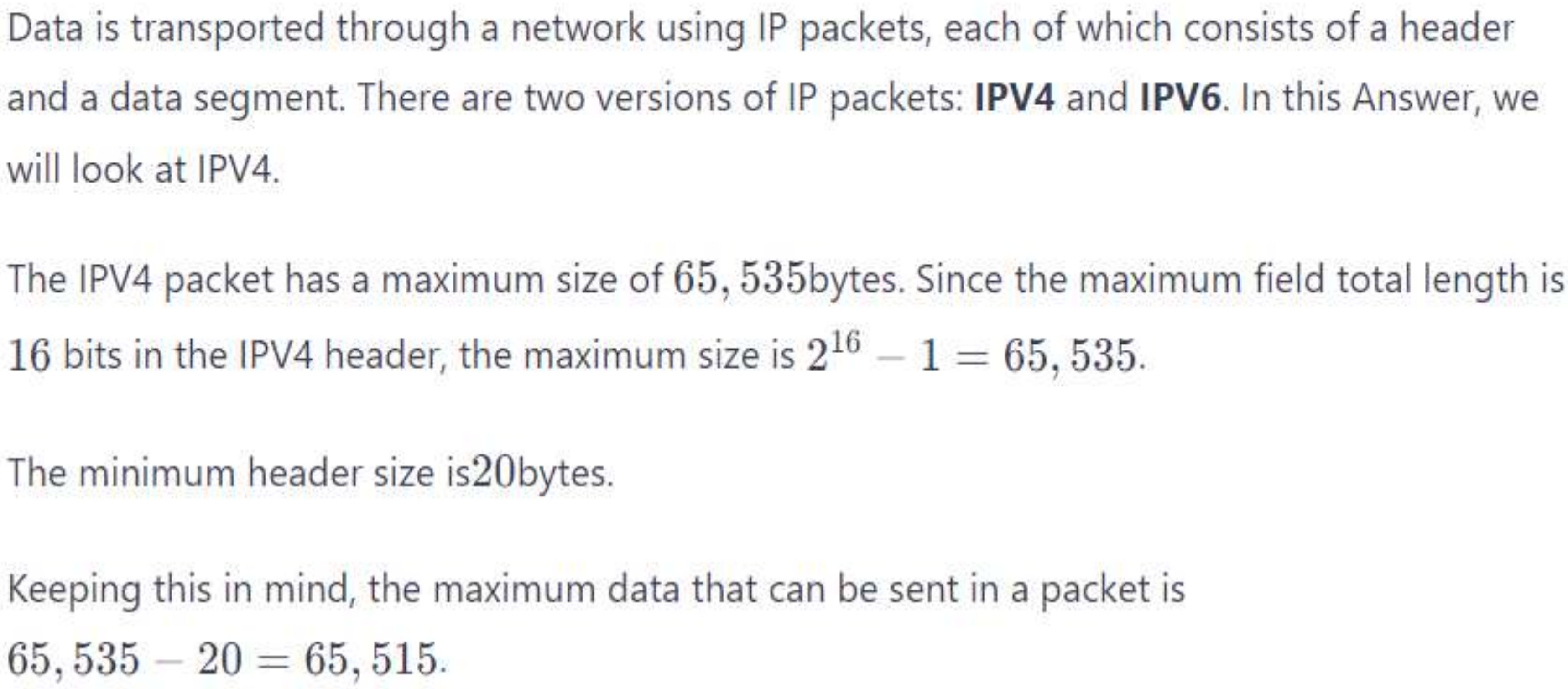
- **Address Format**: 32-bit address
- **Address Notation**: Dotted decimal format (e.g., 192.168.1.1)
- **Total Addresses**: Approximately **4.3 billion (2^{32})**
- **Header Size**: Typically 20 bytes
- **Address Classes**: Divided into five classes (A, B, C, D, and E)
- **Subnetting**: Supports subnetting and classless addressing (CIDR)
- **Broadcasting**: Supports broadcast communication
- **Security**: Limited built-in security; relies on additional security mechanisms
- **Address Depletion**: Due to the growing number of internet-connected devices, IPv4 faces exhaustion issues.

IP fragmentation, reassembly

MTU

- **MTU (Maximum Transmission Unit)**: It is the maximum size of a packet that can be transmitted over the network.

How do IP fragmentation and reassembly work?



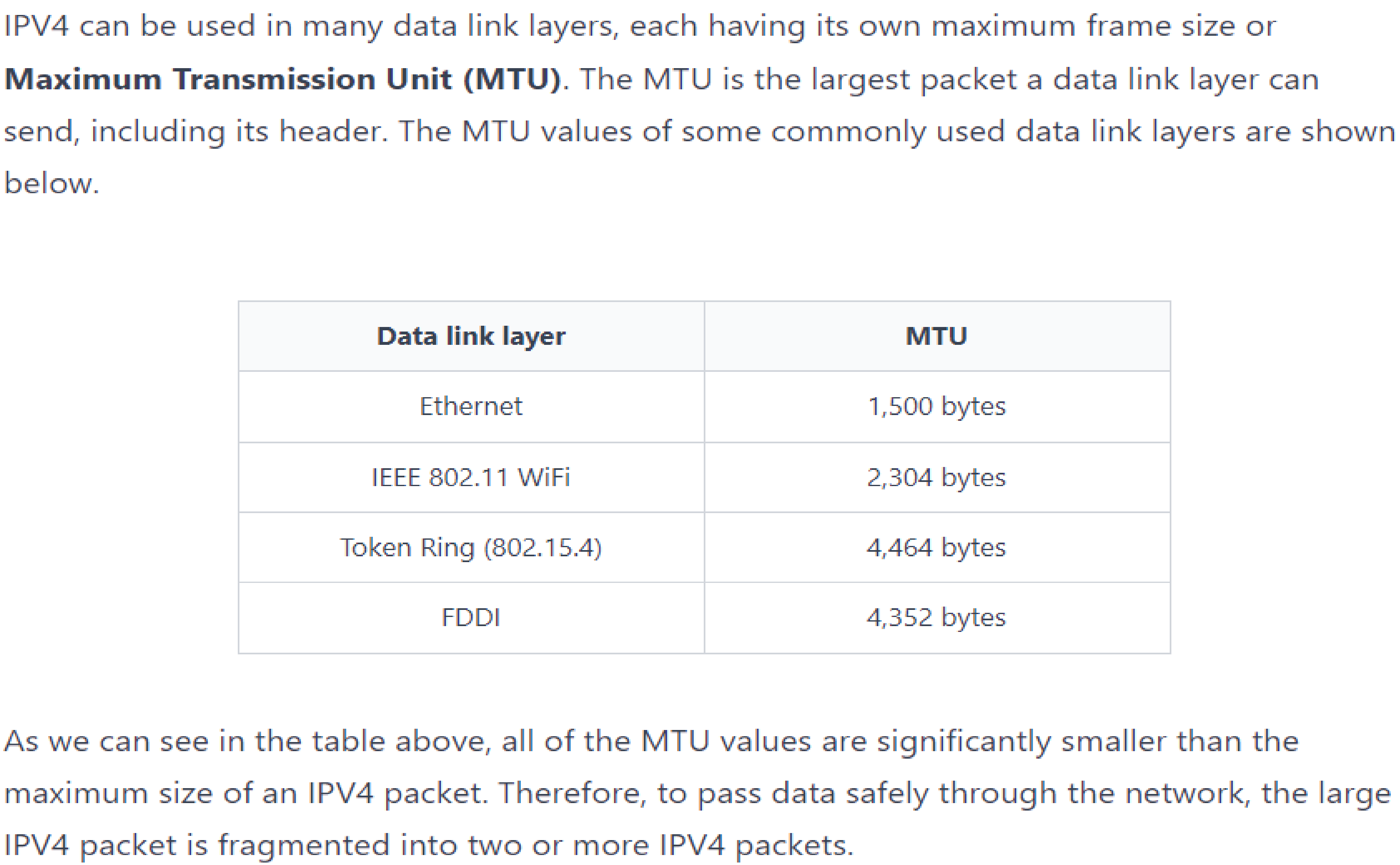
Data is transported through a network using IP packets, each of which consists of a header and a data segment. There are two versions of IP packets: **IPV4** and **IPV6**. In this Answer, we will look at IPV4.

The IPV4 packet has a maximum size of $65,535$bytes. Since the maximum field total length is 16 bits in the IPV4 header, the maximum size is $2^{16} - 1 = 65,535$.

The minimum header size is20bytes.

Keeping this in mind, the maximum data that can be sent in a packet is $65,535 - 20 = 65,515$.

Why is fragmentation needed?



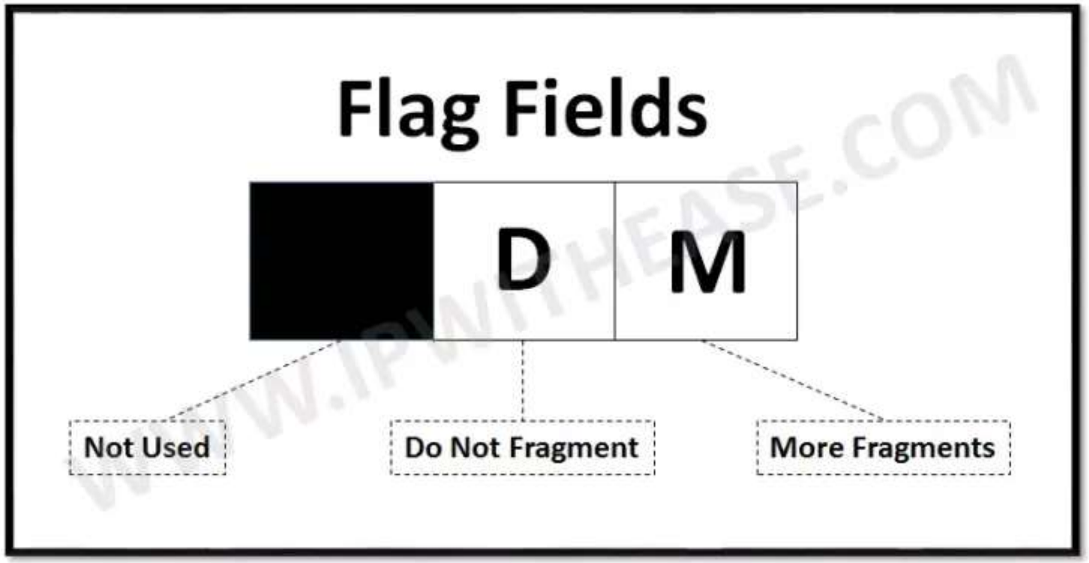
IPV4 can be used in many data link layers, each having its own maximum frame size or **Maximum Transmission Unit (MTU)**. The MTU is the largest packet a data link layer can send, including its header. The MTU values of some commonly used data link layers are shown below.

Data link layer	MTU
Ethernet	1,500 bytes
IEEE 802.11 WiFi	2,304 bytes
Token Ring (802.15.4)	4,464 bytes
FDDI	4,352 bytes

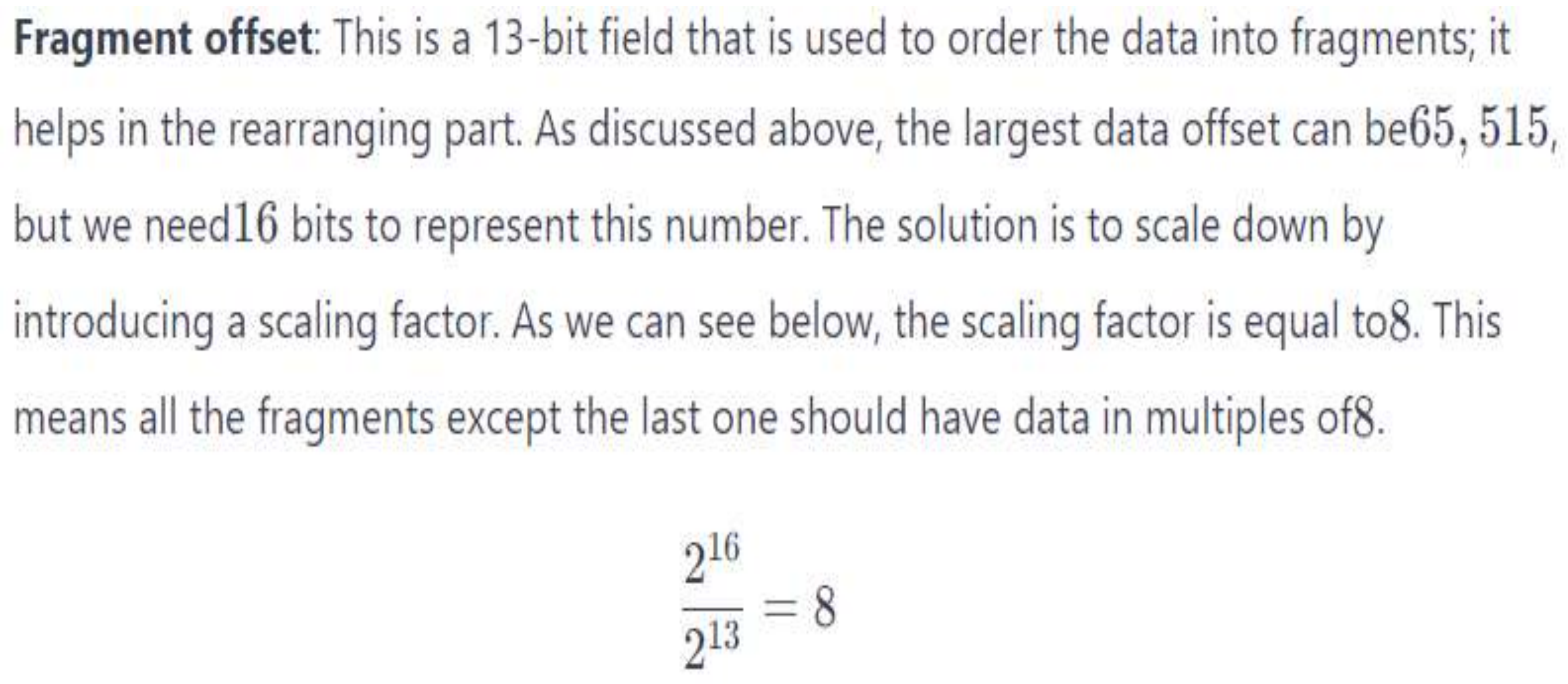
As we can see in the table above, all of the MTU values are significantly smaller than the maximum size of an IPV4 packet. Therefore, to pass data safely through the network, the large IPV4 packet is fragmented into two or more IPV4 packets.

Fragment Offset

- **Total Length Field**: After fragmenting, this field indicates the length of each fragment, not the length of the overall message.
- *Normally, the fragment size is selected to match the MTU value in bytes after subtracting the IP header size of 20 bytes or more.*
- **Identification Number**: All the fragments of the same packet have the same identification number to allow the receiving device to identify all the fragments of a single packet.
- **Flags**: It is a 3-bit field which is used to identify the fragments.
- **bit 0: Reserved; must be zero**
- **bit 1: Don't Fragment (DF)**
- **bit 2: More Fragments (MF)**
- The MF bit is set for all the fragments except the last one for which it is zero.
- The DF bit is set to disable the fragmentation and in this case, if the packet size is greater than MTU value then it is dropped.



Fragment offset:



Fragment offset: This is a 13-bit field that is used to order the data into fragments; it helps in the rearranging part. As discussed above, the largest data offset can be65, 515, but we need16 bits to represent this number. The solution is to scale down by introducing a scaling factor. As we can see below, the scaling factor is equal to8. This means all the fragments except the last one should have data in multiples of8.

$$\frac{2^{16}}{2^{13}} = 8$$

Note

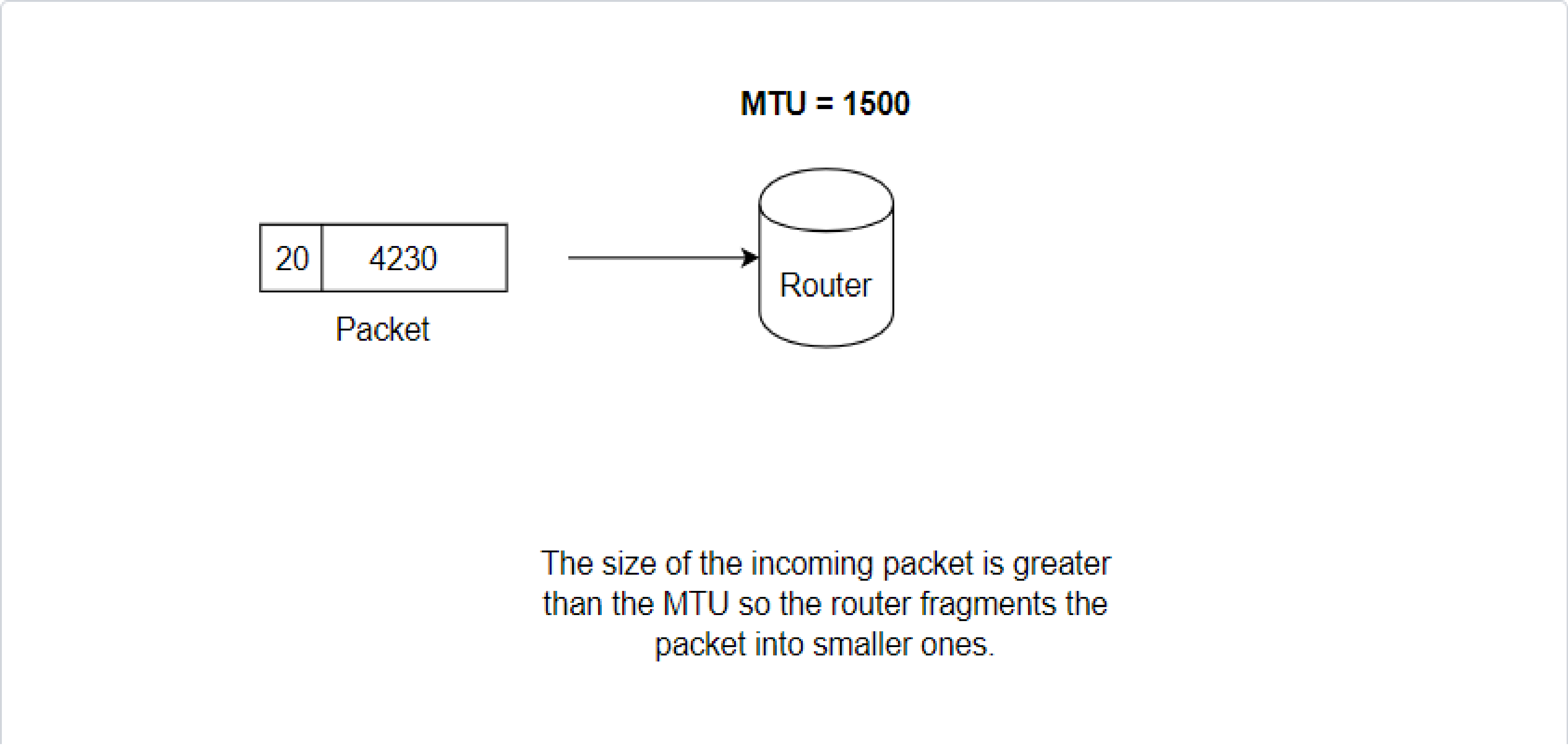
- **The fragmentation offset value for the first fragment is always 0.**

- **The field is 13 bits wide, so the offset can be from 0 to 8191.**

- **Fragments are specified in units of 8 bytes, which is why fragment length must be a multiple of 8.**

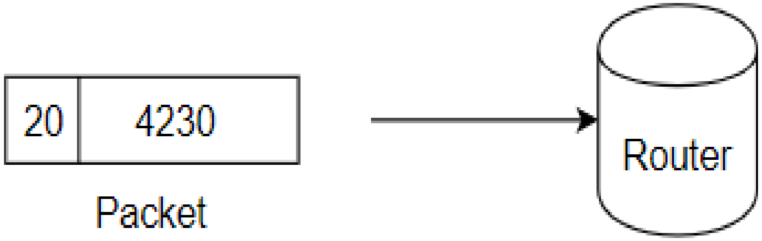
Example

Consider an example where a packet with a size of $4,250$ bytes arrives at an IP router with an MTU of $1,500$. As the packet size is greater than that of the MTU, it needs to be fragmented. The process of fragmentation is shown below:



The size of the incoming packet is greater than the MTU so the router fragments the packet into smaller ones.

MTU = 1500



20 | 4230

Packet

Router

The first fragmented packet contains 1,480 bytes of data as it is divisible by 8, and 1,480 + 20 = 1,500, which is equal to the MTU

Packet 1

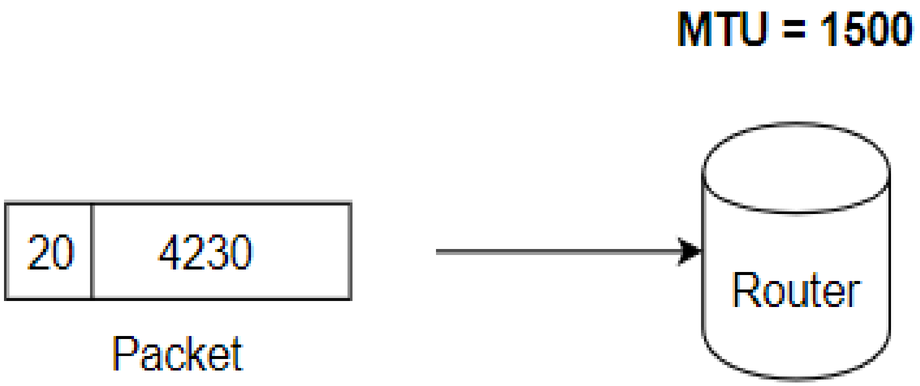
Identification: 231

Fragment offset: 0

MF : 1

Header length : 5

The header length is 5 because it has a scaling factor of 4, meaning 5 x 4 = 20



The second fragmented packet contains 1,480 bytes of data as it is divisible by 8, and 1,480 + 20 = 1,500 which is equal to the MTU

Packet 1

Identification: 231

Fragment offset: 0

MF : 1

Header length : 5

Packet 2

Identification: 231

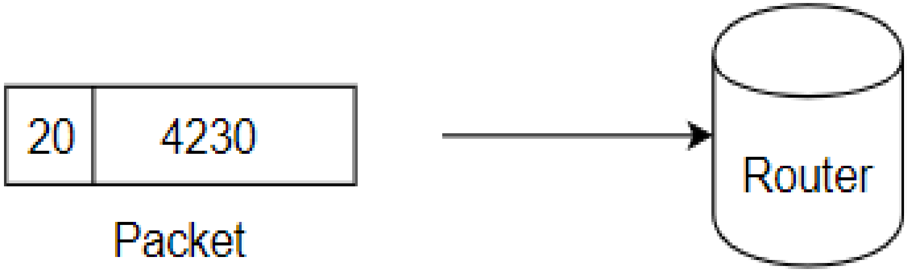
Fragment offset: 185

MF : 1

Header length : 5

The fragment offset is calculated by dividing the data in the first packet by 8 (1,480 / 8 = 185) and then adding it to the offset of the previous packet which is 0, so 185 + 0 = 185

MTU = 1500



20 | 4230

Packet

Router

The third fragmented packet contains the left over data which is 1270 bytes

MF = 0 as it is the last fragment

Packet 1

Identification: 231

Fragment offset: 0

MF : 1

Header length : 5

Packet 2

Identification: 231

Fragment offset: 185

MF : 1

Header length : 5

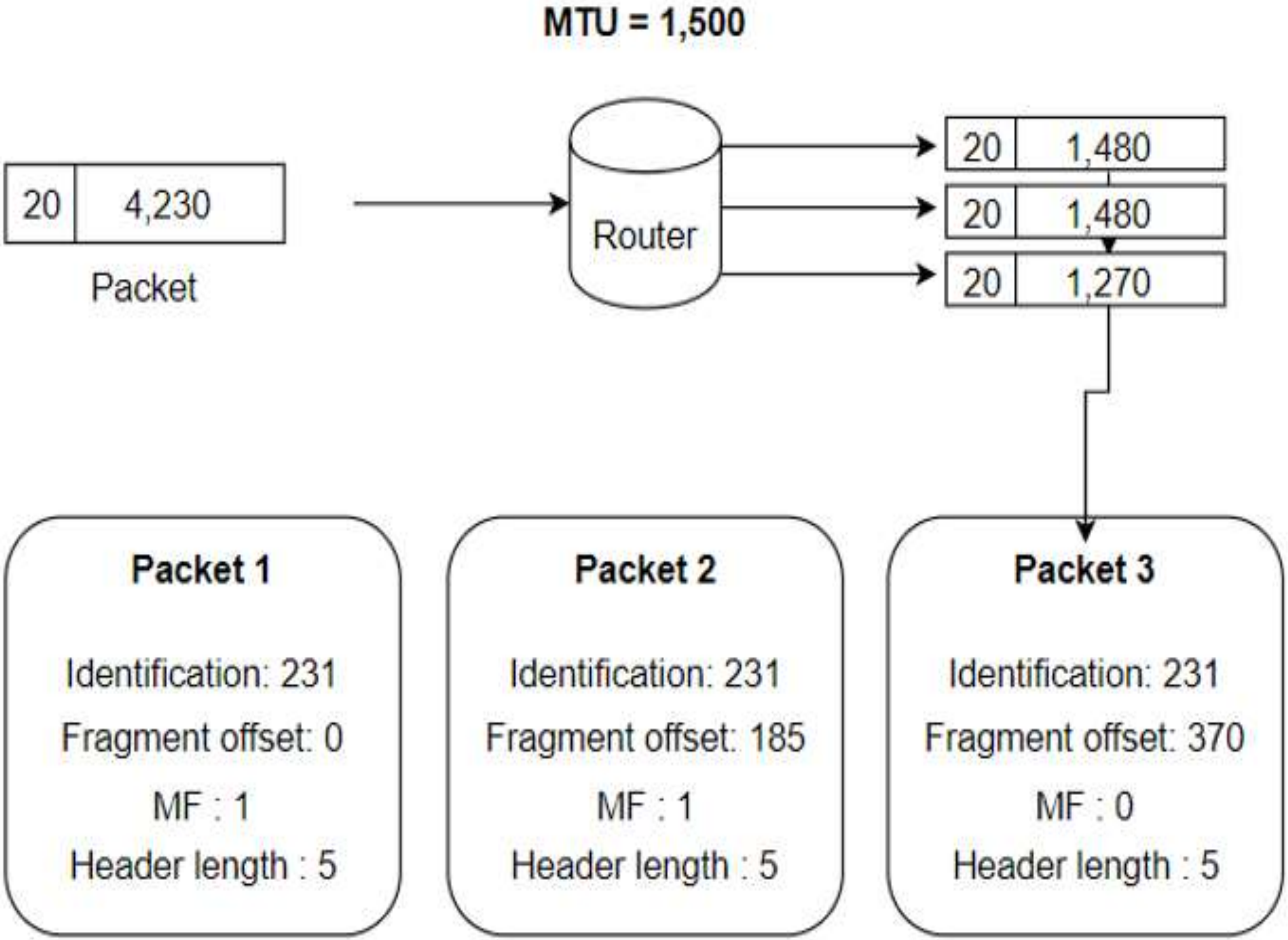
Packet 3

Identification: 231

Fragment offset: 370

MF : 0

Header length : 5



MTU = 1,500

Packet 1

Identification: 231

Fragment offset: 0

MF : 1

Header length : 5

Packet 2

Identification: 231

Fragment offset: 185

MF : 1

Header length : 5

Packet 3

Identification: 231

Fragment offset: 370

MF : 0

Header length : 5

Example

- Let us take an example to understand the calculation for fragmentation offset:

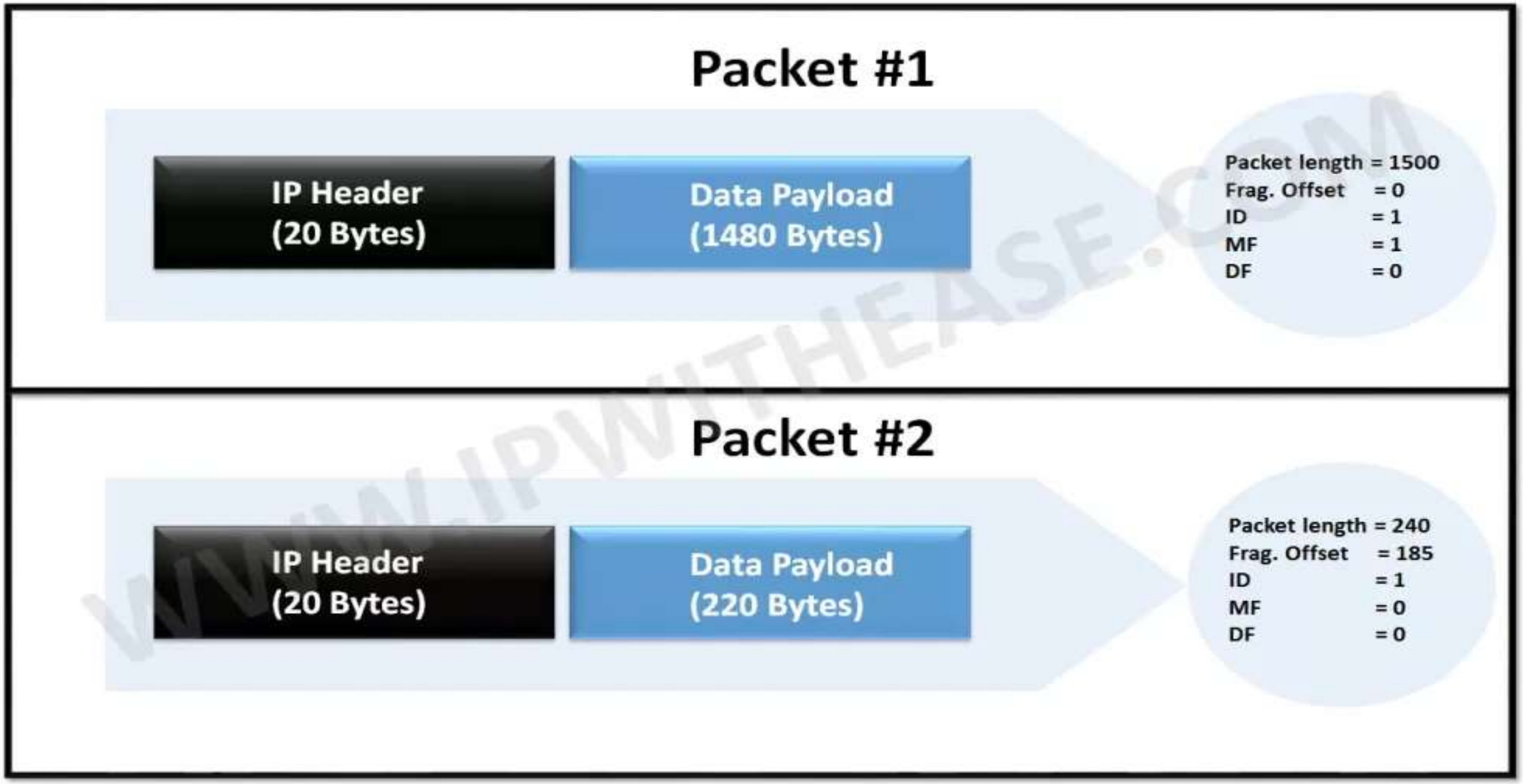
- **Suppose we have a packet of 1700 bytes to be transmitted over an MTU of 1500 bytes.**

First fragment:

- Fragment Offset: 0
- ID : 1
- MF = 1
- DF = 0
- Total Length : 1500 bytes
- Data Payload = 1500 - 20 bytes of IP header = **1480 bytes**

Second Fragment:

- Fragment Offset: 185 (Previous Offset + Previous Fragment Data transmitted/8)
- ID : 1
- MF = 0
- DF = 0
- Total Length: 240 bytes
- Data Payload = 240 bytes – 20 bytes of IP header = **220 bytes**



Packet #1

IP Header
(20 Bytes)

Data Payload
(1480 Bytes)

Packet length = 1500
Frag. Offset = 0
ID = 1
MF = 1
DF = 0

Packet #2

IP Header
(20 Bytes)

Data Payload
(220 Bytes)

Packet length = 240
Frag. Offset = 185
ID = 1
MF = 0
DF = 0

Test Yourself

Given the sample MTU size of 100 and an IP packet of size 999, how many fragments will be created?

a) 11

b) 12

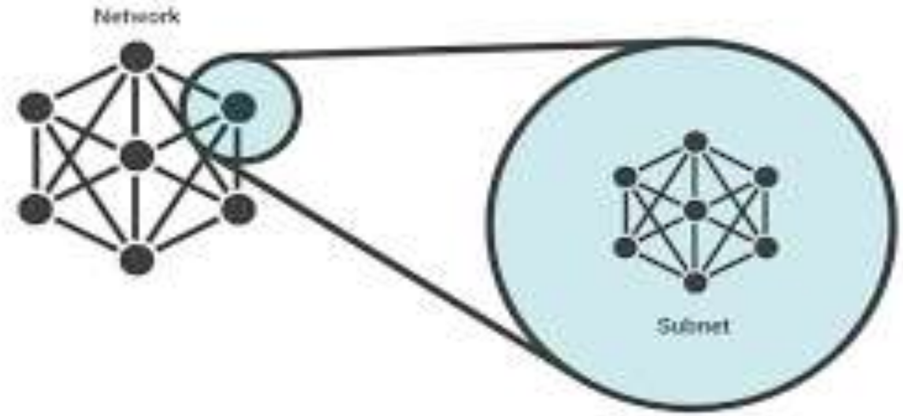
c) 13

Solution

- Maximum Data in Each Fragment:MTU (100 bytes) - Header Size (20 bytes) = 80 bytes of Data
- Total Data Size to Transmit:999 bytes (Total IP Packet Size) - 20 bytes (Initial Header) = 979 bytes of Data
- Number of Fragments:979/80=12.23
- So total **13** fragments are required

What is Subnetting

- **Subnetting** is a process of dividing a large network into smaller, more manageable sub-networks or subnets.
- It helps improve network performance, security, and efficient IP address allocation.



Why subnetting ?

- Efficient IP Address Usage
- Network Security
- Traffic Management
- Easier Troubleshooting
- Reduced Network Congestion

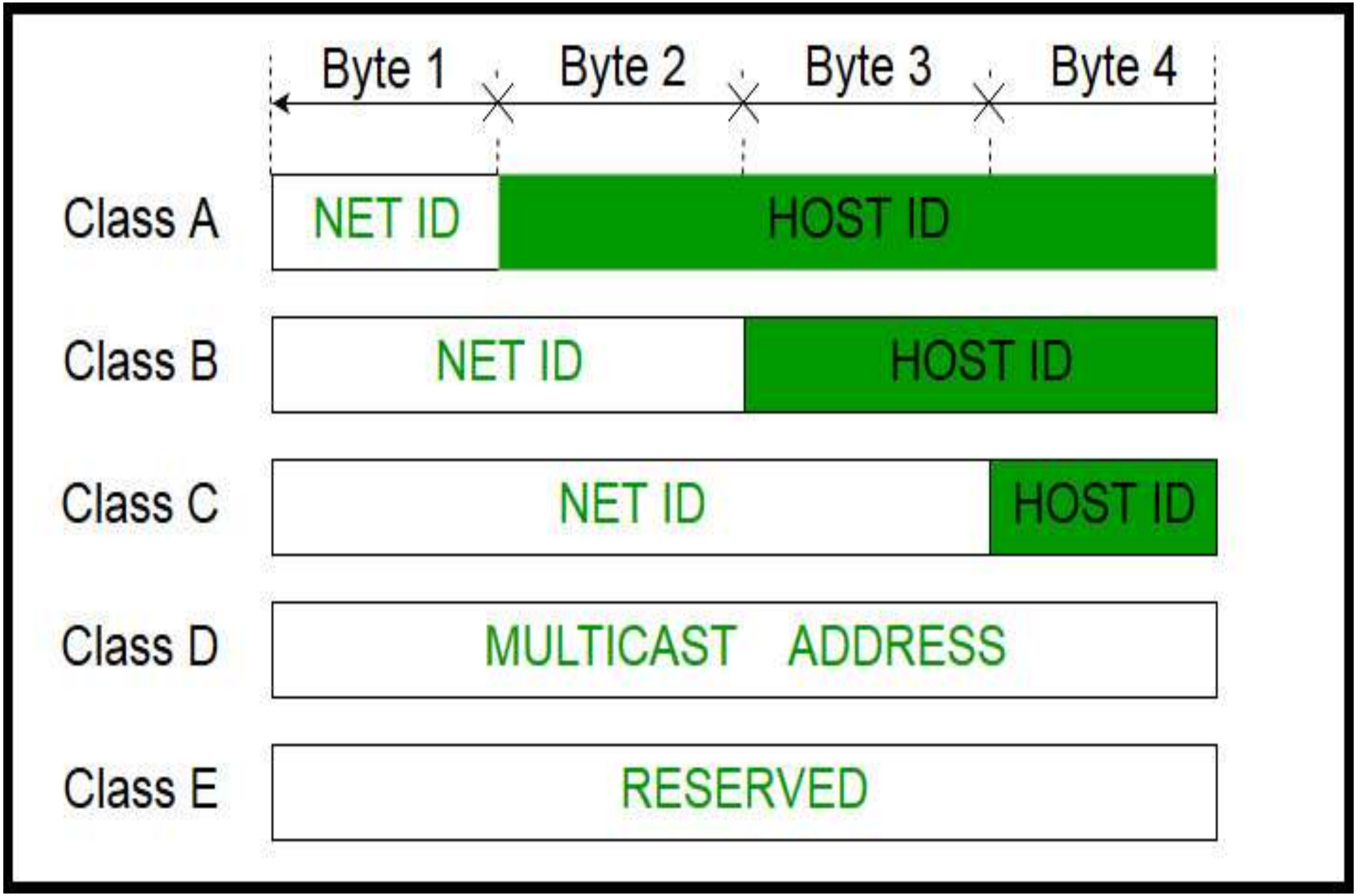
IPv4 Address Structure Recap

- An IPv4 address is a 32-bit number, usually represented in **dotted decimal notation** (e.g., 192.168.1.1).
- IPv4 is divided into:
- **Network Portion** (Identifies the network)
- **Host Portion** (Identifies devices in that network)

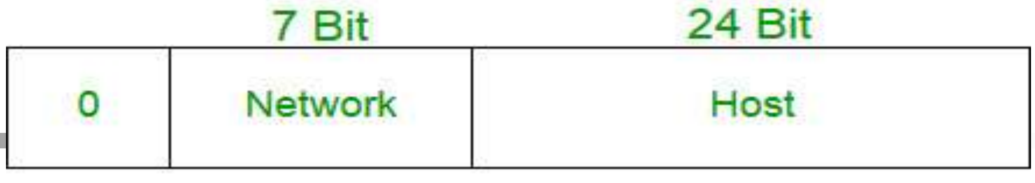
Subnet Mask

- A Subnet Mask defines how many bits are reserved for the network and how many for the host.
- Example:
- IP Address: 192.168.1.0
- Subnet Mask: 255.255.255.0 or /24
- Here, /24 means 24 bits for the network and 8 bits for the host.

Classful Addressing



Class A

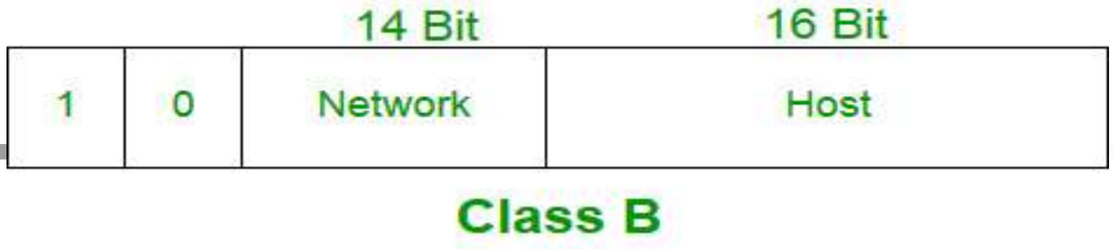


	7 Bit	24 Bit
0	Network	Host

Class A

- The default subnet mask for Class A is 255.0.0.0.
- Therefore, class A has a total of:
- 2^7-2= 126 network ID(Here 2 addresses are subtracted because 0.0.0.0 and 127.x.y.z are special address.)
- 2^24 – 2 = 16,777,214 host ID
- IP addresses belonging to class A ranges from 1.x.x.x – 126.x.x.x

Class B

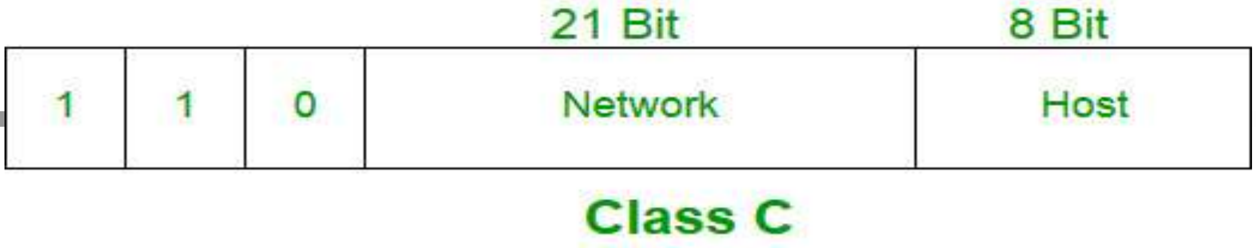


		14 Bit	16 Bit
1	0	Network	Host

Class B

- The default subnet mask for class B is 255.255.x.x.
- Class B has a total of:
  - 2^14 = 16384 network address
  - 2^16 – 2 = 65534 host address
- IP addresses belonging to class B ranges from 128.0.x.x – 191.255.x.x.

Class C

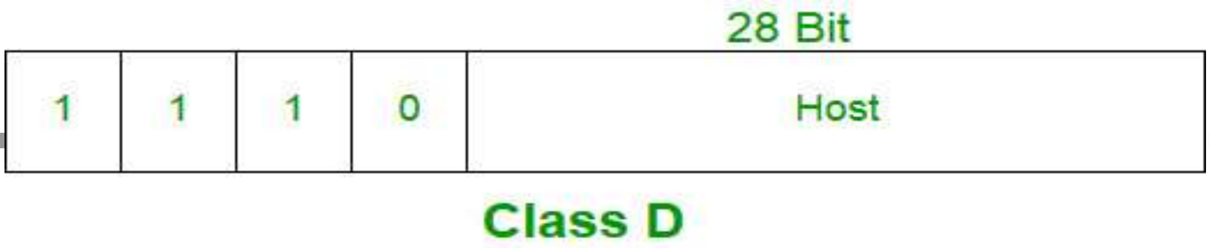


			21 Bit	8 Bit
1	1	0	Network	Host

Class C

- The default subnet mask for class C is 255.255.255.x.
- Class C has a total of:
  - 2^21 = 2097152 network address
  - 2^8 – 2 = 254 host address
- IP addresses belonging to class C range from 192.0.0.x – 223.255.255.x.

Class D



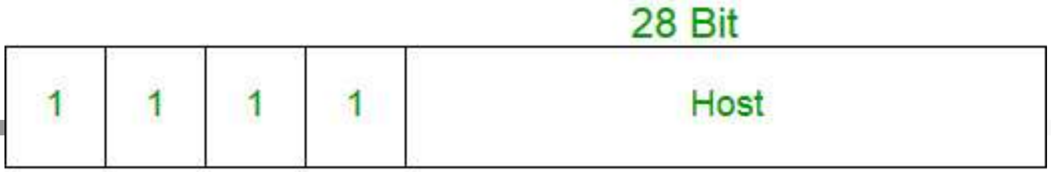
28 Bit

1	1	1	0	Host

Class D

- IP address belonging to class D is reserved for multi-casting.
- The higher-order bits of the first octet of IP addresses belonging to class D is always set to 1110.
- The remaining bits are for the address that interested hosts recognize.
- Class D does not possess any subnet mask.
- IP addresses belonging to class D range from 224.0.0.0 – 239.255.255.255.

Class E

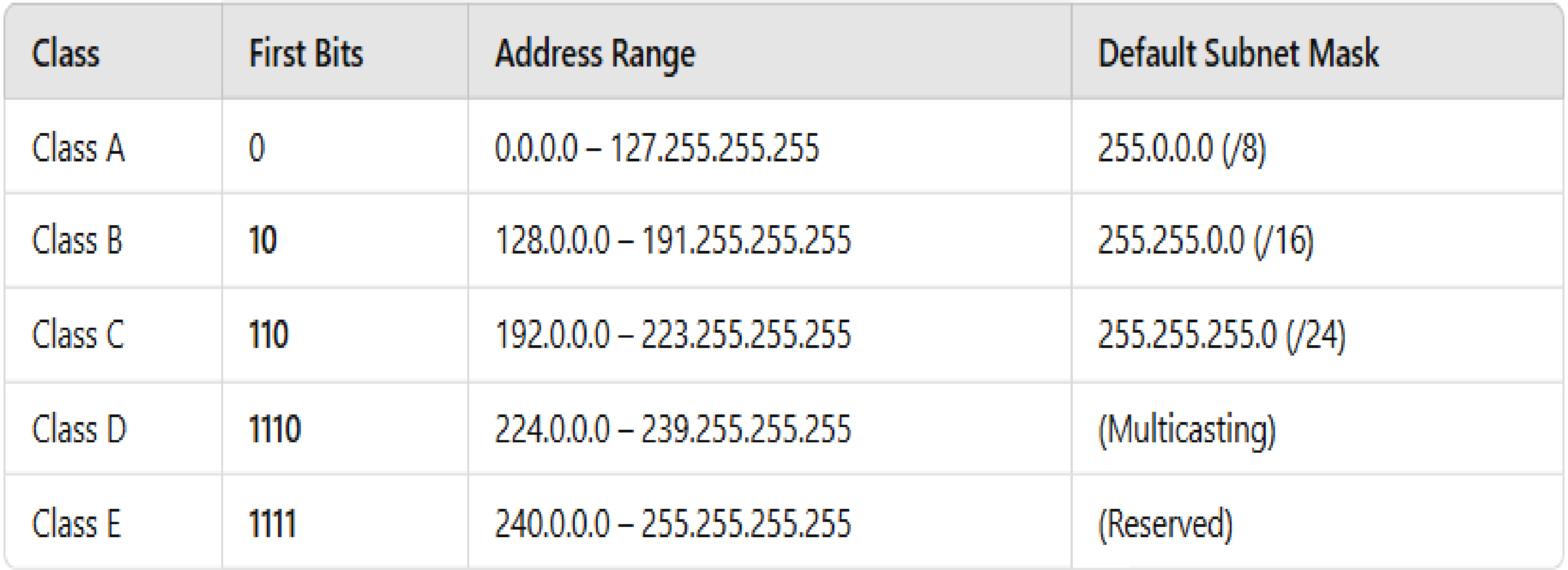


28 Bit				
1	1	1	1	Host

Class E

- IP addresses belonging to class E are reserved for experimental and research purposes.
- IP addresses of class E range from 240.0.0.0 – 255.255.255.254.
- This class doesn’t have any subnet mask.
- The higher-order bits of the first octet of class E are always set to 1111.

IP Address Classes



Class	First Bits	Address Range	Default Subnet Mask
Class A	0	0.0.0.0 – 127.255.255.255	255.0.0.0 (/8)
Class B	10	128.0.0.0 – 191.255.255.255	255.255.0.0 (/16)
Class C	110	192.0.0.0 – 223.255.255.255	255.255.255.0 (/24)
Class D	1110	224.0.0.0 – 239.255.255.255	(Multicasting)
Class E	1111	240.0.0.0 – 255.255.255.255	(Reserved)

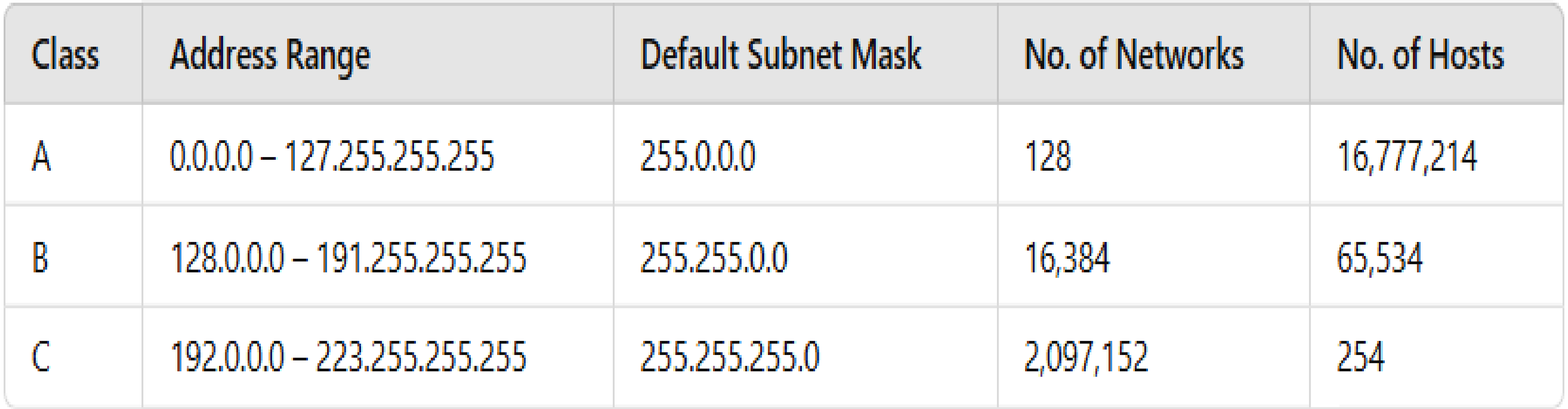
Subnetting Formula

- Number of Subnets: 2^n
- Number of Hosts: $2^h - 2$

Where

  - **n** = Number of borrowed bits
  - **h** = Remaining host bits

Classful IP Addressing



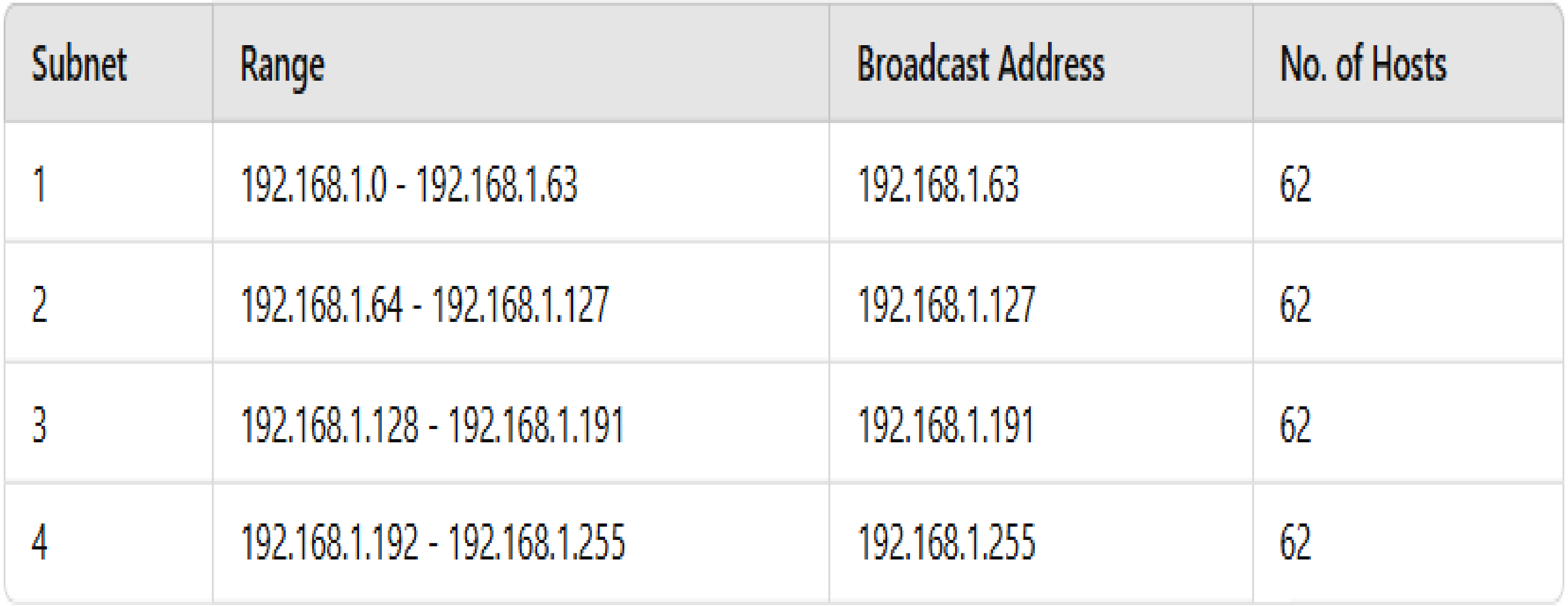
Class	Address Range	Default Subnet Mask	No. of Networks	No. of Hosts
A	0.0.0.0 – 127.255.255.255	255.0.0.0	128	16,777,214
B	128.0.0.0 – 191.255.255.255	255.255.0.0	16,384	65,534
C	192.0.0.0 – 223.255.255.255	255.255.255.0	2,097,152	254

Classless Inter-Domain Routing (CIDR)

- CIDR is a method to allocate IP addresses more flexibly than classful addressing.
- Example:
- 192.168.1.0/26 means 26 bits for the network, leaving 6 bits for hosts.

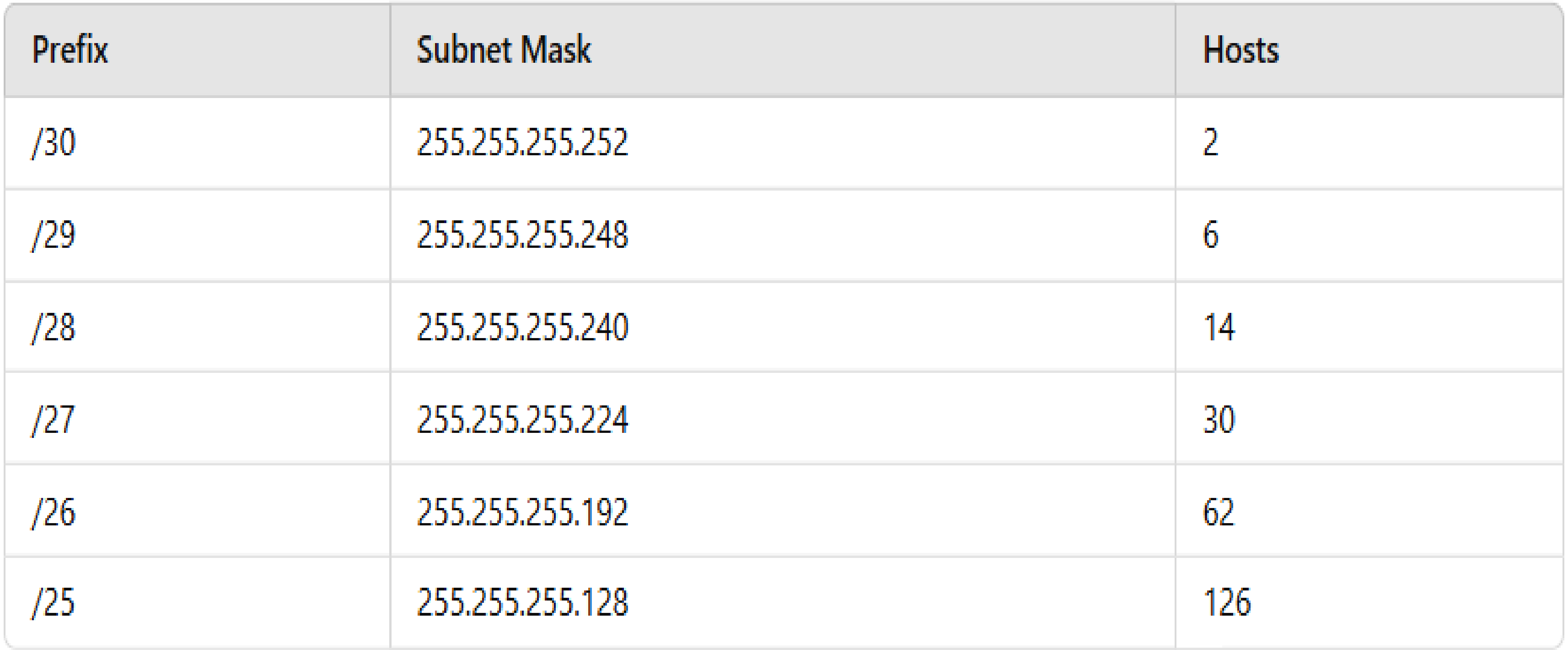
Example

- **IP Address:** 192.168.1.0/26
  Subnet Mask: 255.255.255.192



Subnet	Range	Broadcast Address	No. of Hosts
1	192.168.1.0 - 192.168.1.63	192.168.1.63	62
2	192.168.1.64 - 192.168.1.127	192.168.1.127	62
3	192.168.1.128 - 192.168.1.191	192.168.1.191	62
4	192.168.1.192 - 192.168.1.255	192.168.1.255	62

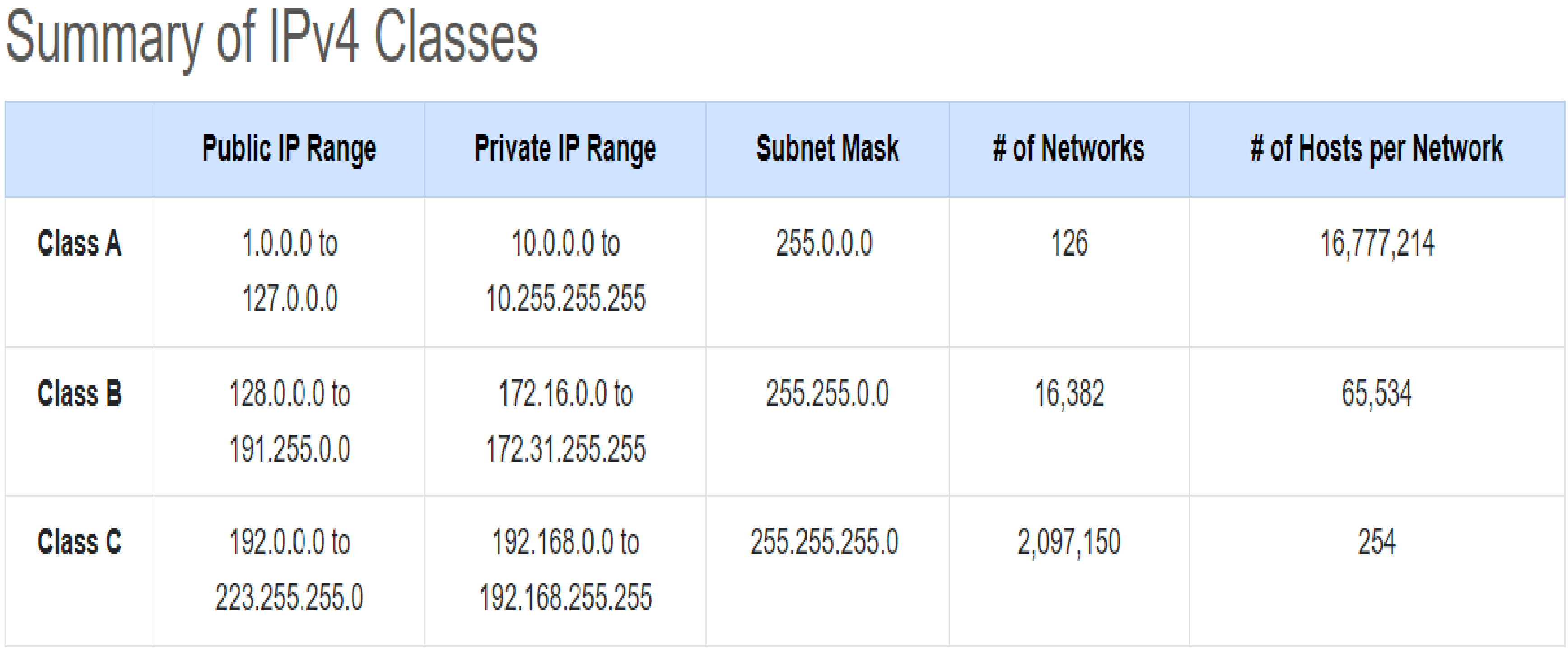
Subnetting Tricks



Prefix	Subnet Mask	Hosts
/30	255.255.255.252	2
/29	255.255.255.248	6
/28	255.255.255.240	14
/27	255.255.255.224	30
/26	255.255.255.192	62
/25	255.255.255.128	126

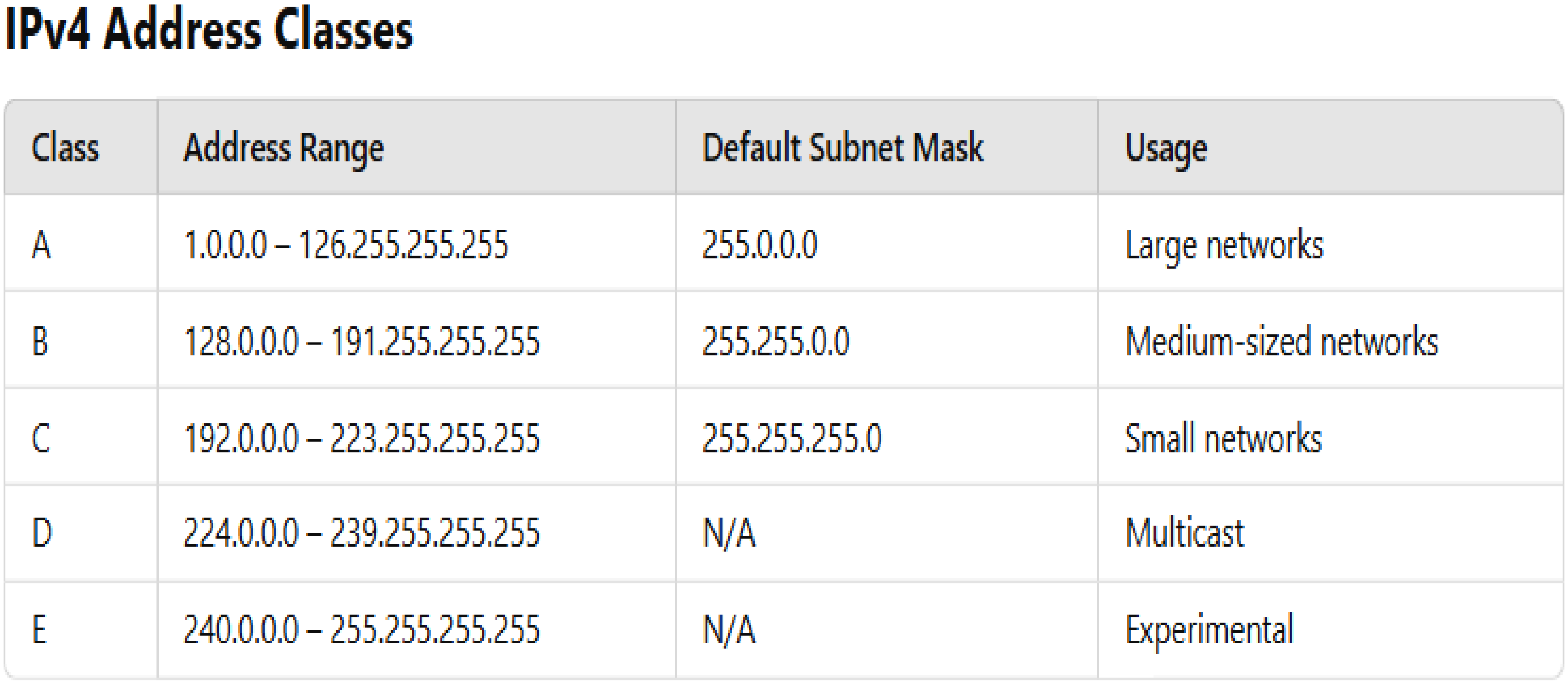
How to Calculate Subnets?

1. Convert Subnet Mask to Binary.
2. Identify Borrowed Bits.
3. Calculate Number of Subnets.
4. Calculate Host Range.
5. Assign Network Address, Broadcast Address, and Host Range.



Summary of IPv4 Classes

	Public IP Range	Private IP Range	Subnet Mask	# of Networks	# of Hosts per Network
Class A	1.0.0.0 to 127.0.0.0	10.0.0.0 to 10.255.255.255	255.0.0.0	126	16,777,214
Class B	128.0.0.0 to 191.255.0.0	172.16.0.0 to 172.31.255.255	255.255.0.0	16,382	65,534
Class C	192.0.0.0 to 223.255.255.0	192.168.0.0 to 192.168.255.255	255.255.255.0	2,097,150	254



IPv4 Address Classes

Class	Address Range	Default Subnet Mask	Usage
A	1.0.0.0 – 126.255.255.255	255.0.0.0	Large networks
B	128.0.0.0 – 191.255.255.255	255.255.0.0	Medium-sized networks
C	192.0.0.0 – 223.255.255.255	255.255.255.0	Small networks
D	224.0.0.0 – 239.255.255.255	N/A	Multicast
E	240.0.0.0 – 255.255.255.255	N/A	Experimental

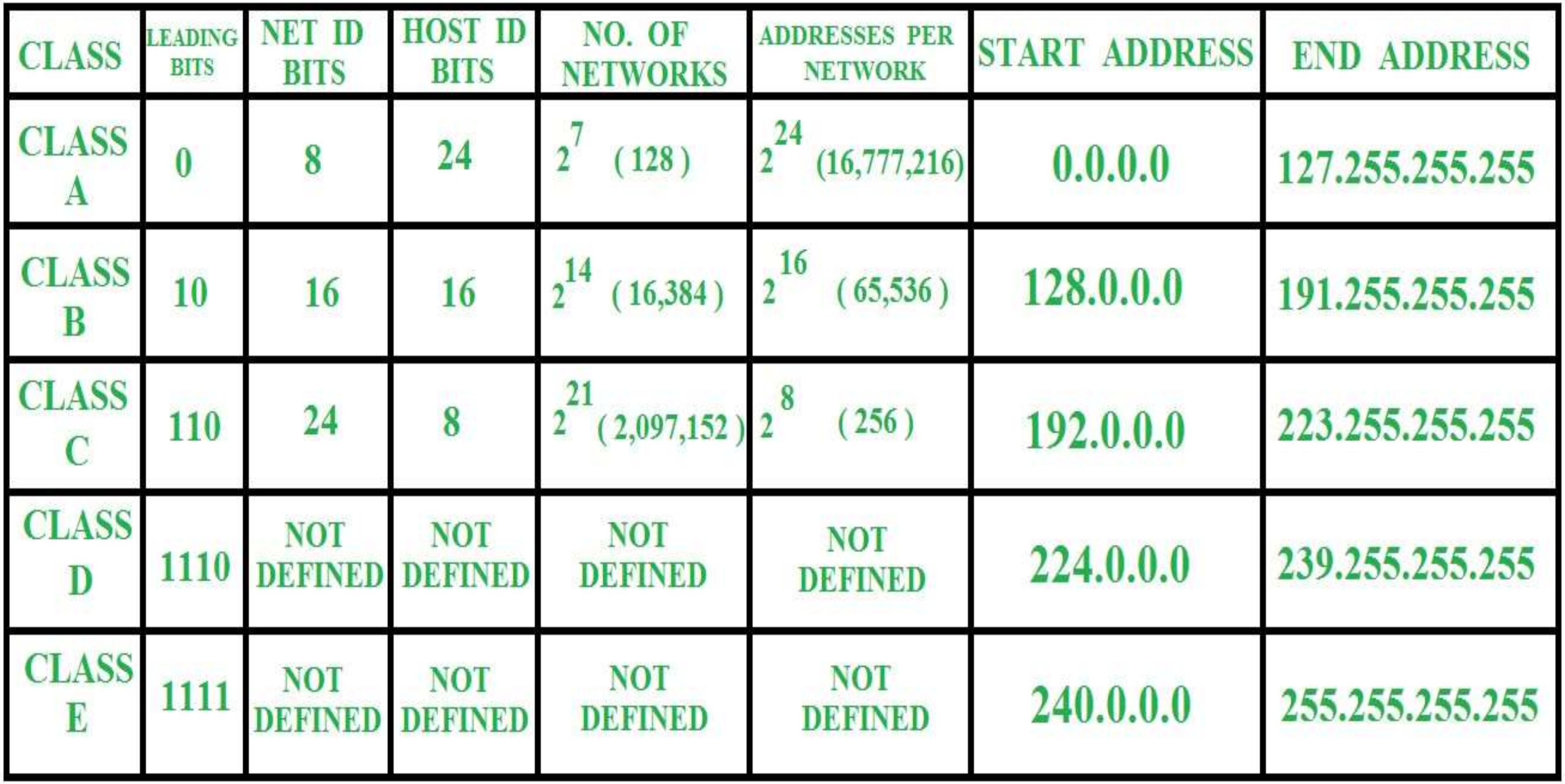
Range of Special IP Addresses

- 169.254.0.0 – 169.254.0.16 : Link-local addresses
- 127.0.0.0 – 127.0.0.8 : Loop-back addresses
- 0.0.0.0 – 0.0.0.8: used to communicate within the current network.

Private IP addresses

- *Private IP addresses are IP addresses reserved for use within private networks and are not directly accessible from the internet.*
- *They are used to allow devices within a private network to communicate with each other.*
- Class A: 10.0. 0.0 to 10.255. 255.255.
- Class B: 172.16. 0.0 to 172.31. 255.255.
- Class C: 192.168. 0.0 to 192.168. 255.255.

Summary



CLASS	LEADING BITS	NET ID BITS	HOST ID BITS	NO. OF NETWORKS	ADDRESSES PER NETWORK	START ADDRESS	END ADDRESS
CLASS A	0	8	24	2^7 (128)	2^{24} (16,777,216)	0.0.0.0	127.255.255.255
CLASS B	10	16	16	2^{14} (16,384)	2^{16} (65,536)	128.0.0.0	191.255.255.255
CLASS C	110	24	8	2^{21} (2,097,152)	2^8 (256)	192.0.0.0	223.255.255.255
CLASS D	1110	NOT DEFINED	NOT DEFINED	NOT DEFINED	NOT DEFINED	224.0.0.0	239.255.255.255
CLASS E	1111	NOT DEFINED	NOT DEFINED	NOT DEFINED	NOT DEFINED	240.0.0.0	255.255.255.255

Solve

- Divide the IP Address **172.16.0.0/20** into subnets.

Solution:

- Subnet Mask: 255.255.240.0
- Borrowed Bits: 4
- Number of Subnets: $2^4=16$
- Hosts per Subnet: $2^{12} - 2 = 4094$

IPv6

- An IPv6 address is 128 bits in length and consists of eight, 16-bit fields, with each field bounded by a colon.
- Each field must contain a hexadecimal number
- IPv6 was developed to overcome the limitations of IPv4, mainly addressing exhaustion and improved security.

IPv6 Address Representation

- Full Notation:
  2001:0db8:0000:0000:0000:ff00:0042:8329
- Compressed Notation:
  2001:db8::ff00:42:8329

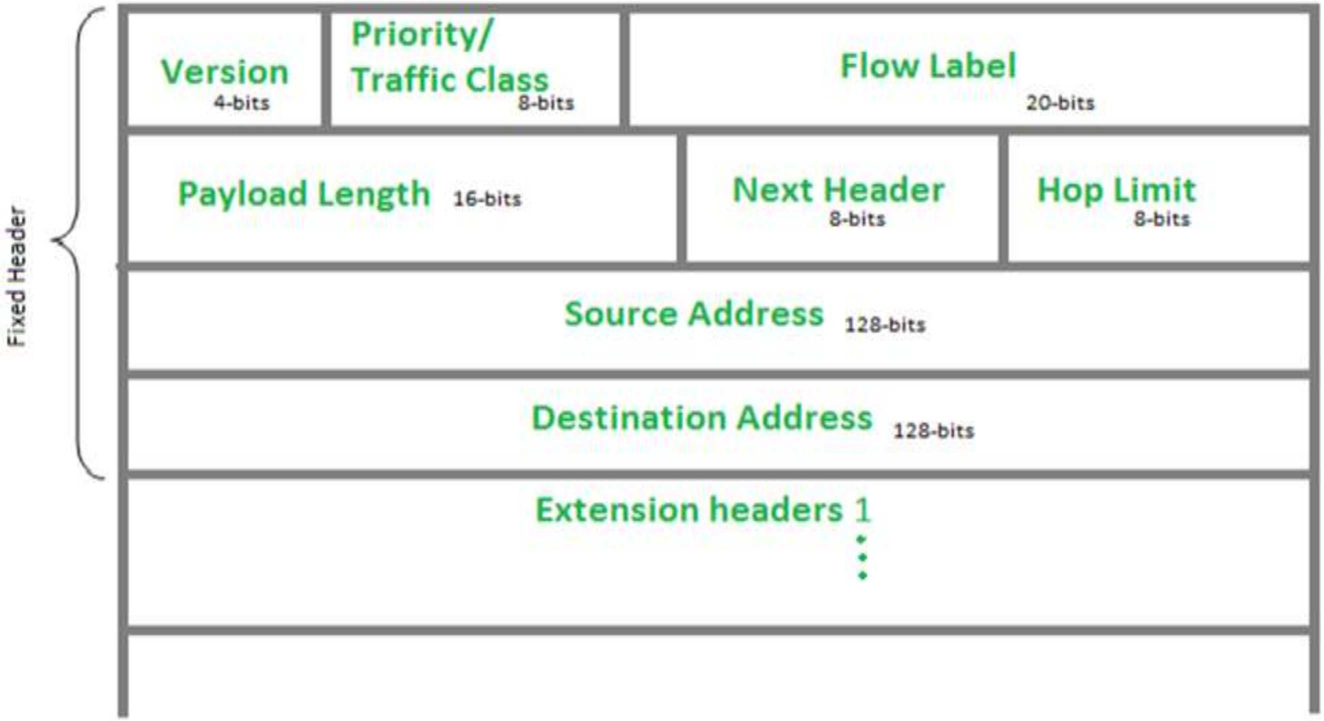
(removes leading zeros and consecutive zeros)

Key Features of IPv6

- **Address Format**: 128-bit address
- **Address Notation**: Hexadecimal notation with colons (e.g., 2001:0db8:85a3::8a2e:0370:7334)
- **Total Addresses**: **Approximately 340 undecillion (2^{128})**
- **Header Size**: 40 bytes (simplified compared to IPv4)
- **Addressing**: No need for subnetting; uses hierarchical addressing
- **Broadcasting**: Does not support broadcast; uses multicast and anycast instead
- **Security**: Built-in security with IPsec (mandatory)
- **Address Configuration**: Supports both stateful (DHCPv6) and stateless (SLAAC) address auto-configuration
- **Backward Compatibility**: IPv4-to-IPv6 transition mechanisms like dual-stack and tunneling

IPv6 datagram format

priority: identify priority among datagrams in flow
flow Label: identify datagrams in same "flow."
next header: identify upper layer protocol for data



Version 4-bits

Priority/ Traffic Class 8-bits

Flow Label 20-bits

Payload Length 16-bits

Next Header 8-bits

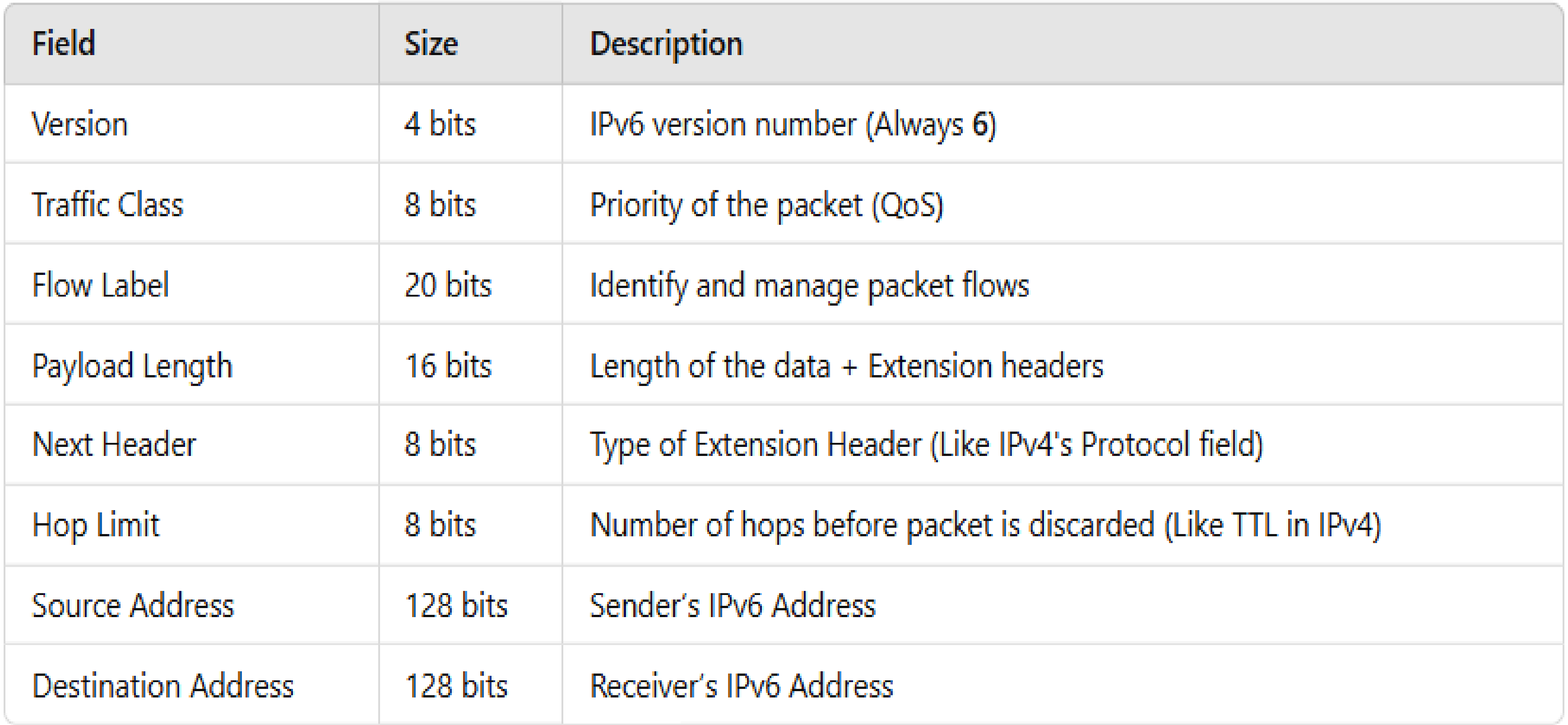
Hop Limit 8-bits

Source Address 128-bits

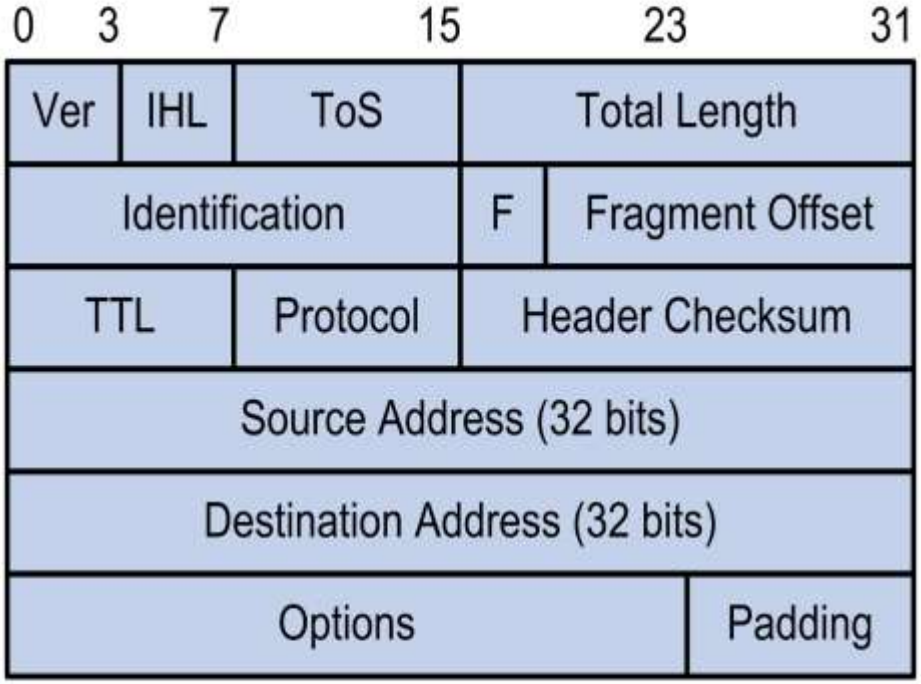
Destination Address 128-bits

Extension headers 1
⋮

IPv6 Header Format (40 bytes)



Field	Size	Description
Version	4 bits	IPv6 version number (Always **6**)
Traffic Class	8 bits	Priority of the packet (QoS)
Flow Label	20 bits	Identify and manage packet flows
Payload Length	16 bits	Length of the data + Extension headers
Next Header	8 bits	Type of Extension Header (Like IPv4's Protocol field)
Hop Limit	8 bits	Number of hops before packet is discarded (Like TTL in IPv4)
Source Address	128 bits	Sender's IPv6 Address
Destination Address	128 bits	Receiver's IPv6 Address



0 3 7 15 23 31

Ver | IHL | ToS | Total Length

Identification | F | Fragment Offset

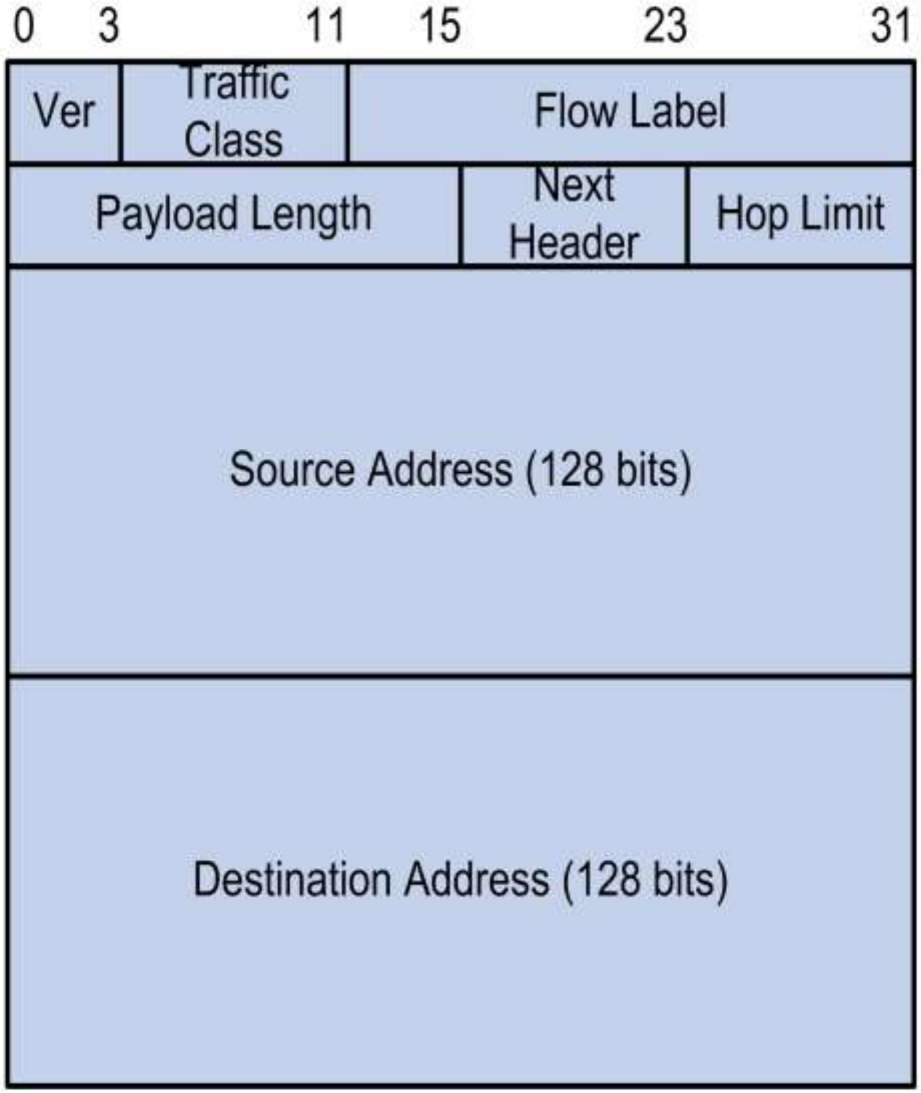
TTL | Protocol | Header Checksum

Source Address (32 bits)

Destination Address (32 bits)

Options | Padding

IPv4 header



0 3 11 15 23 31

Ver | Traffic Class | Flow Label

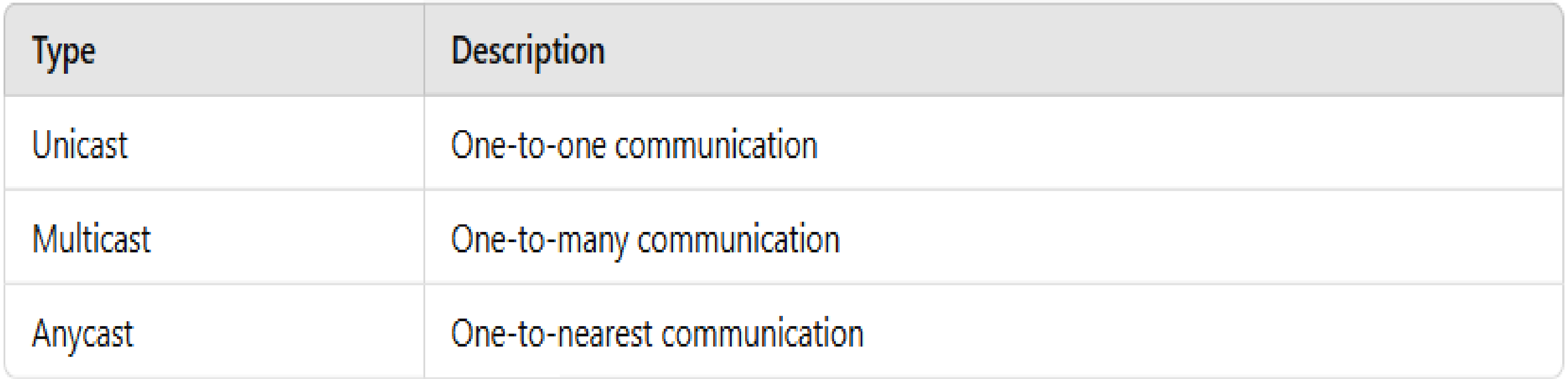
Payload Length | Next Header | Hop Limit

Source Address (128 bits)

Destination Address (128 bits)

Basic IPv6 header

IPv6 Address Types



Type	Description
Unicast	One-to-one communication
Multicast	One-to-many communication
Anycast	One-to-nearest communication

Solve

Given an IPv6 packet with:

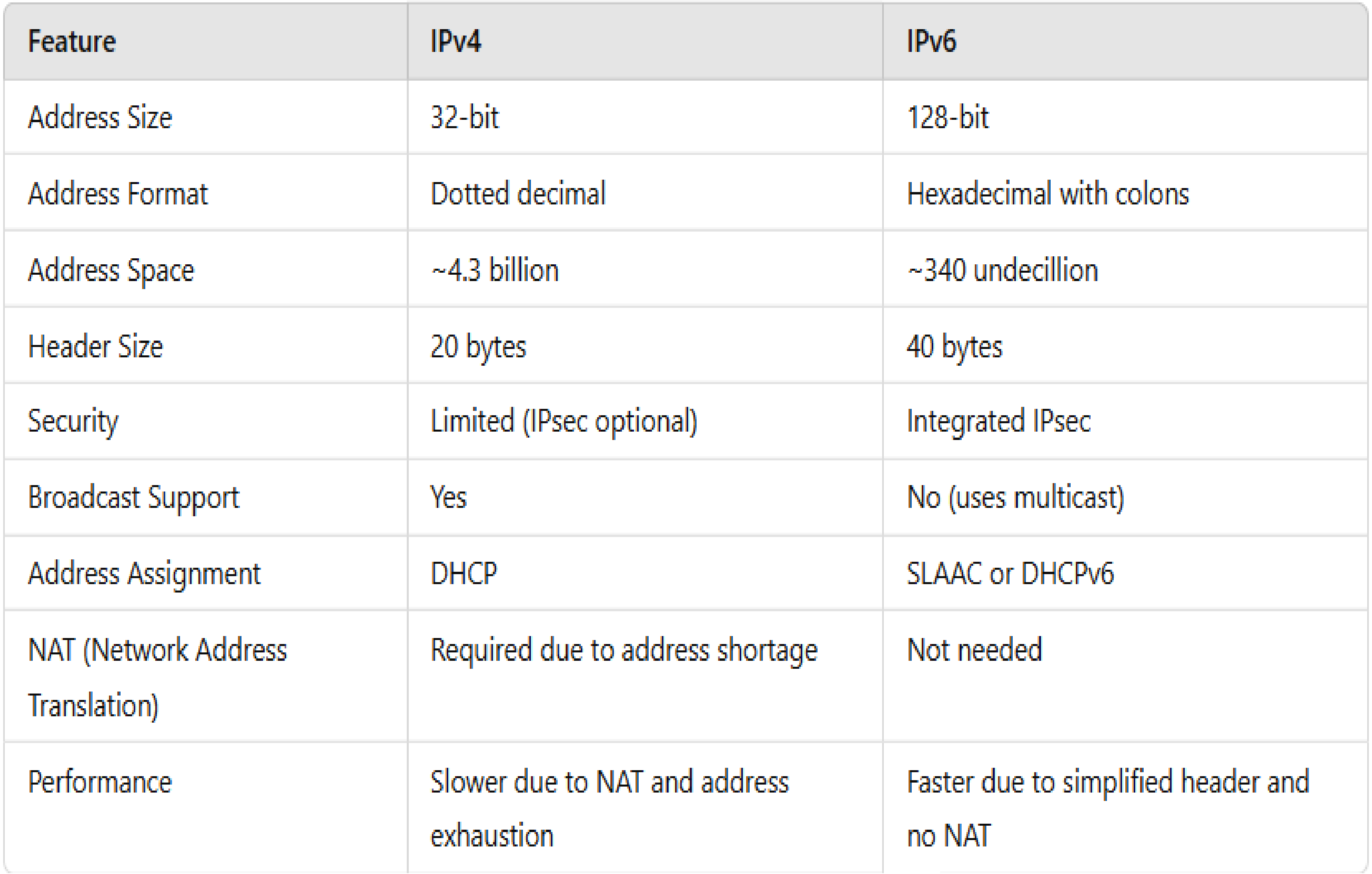
- Payload Length: 1000 bytes
- Hop Limit: 64
- Source Address: 2001:db8::1
- Destination Address: 2001:db8::2

What will be the total IPv6 Packet size?

Solution

- Total Size = Header Size + Payload Length
  = 40 bytes + 1000 bytes
  = **1040 bytes**

Comparison of IPv4 and IPv6



Feature	IPv4	IPv6
Address Size	32-bit	128-bit
Address Format	Dotted decimal	Hexadecimal with colons
Address Space	~4.3 billion	~340 undecillion
Header Size	20 bytes	40 bytes
Security	Limited (IPsec optional)	Integrated IPsec
Broadcast Support	Yes	No (uses multicast)
Address Assignment	DHCP	SLAAC or DHCPv6
NAT (Network Address Translation)	Required due to address shortage	Not needed
Performance	Slower due to NAT and address exhaustion	Faster due to simplified header and no NAT

IP addressing...

Q: how does an ISP get block of addresses?

A: ICANN: Internet Corporation for Assigned Names and Numbers

http://www.icann.org/

- allocates addresses
- manages DNS
- assigns domain names, resolves disputes

UNIT – III

- **Network Layer:** Switching, Logical addressing - IPV4, IPV6; Address mapping - ARP, RARP, BOOTP and DHCP-Delivery, Forwarding and Unicast Routing protocols
- **Transport Layer:** Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), SCTP Congestion Control; Quality of Service (QoS), QoS improving techniques - Leaky Bucket and Token Bucket algorithms

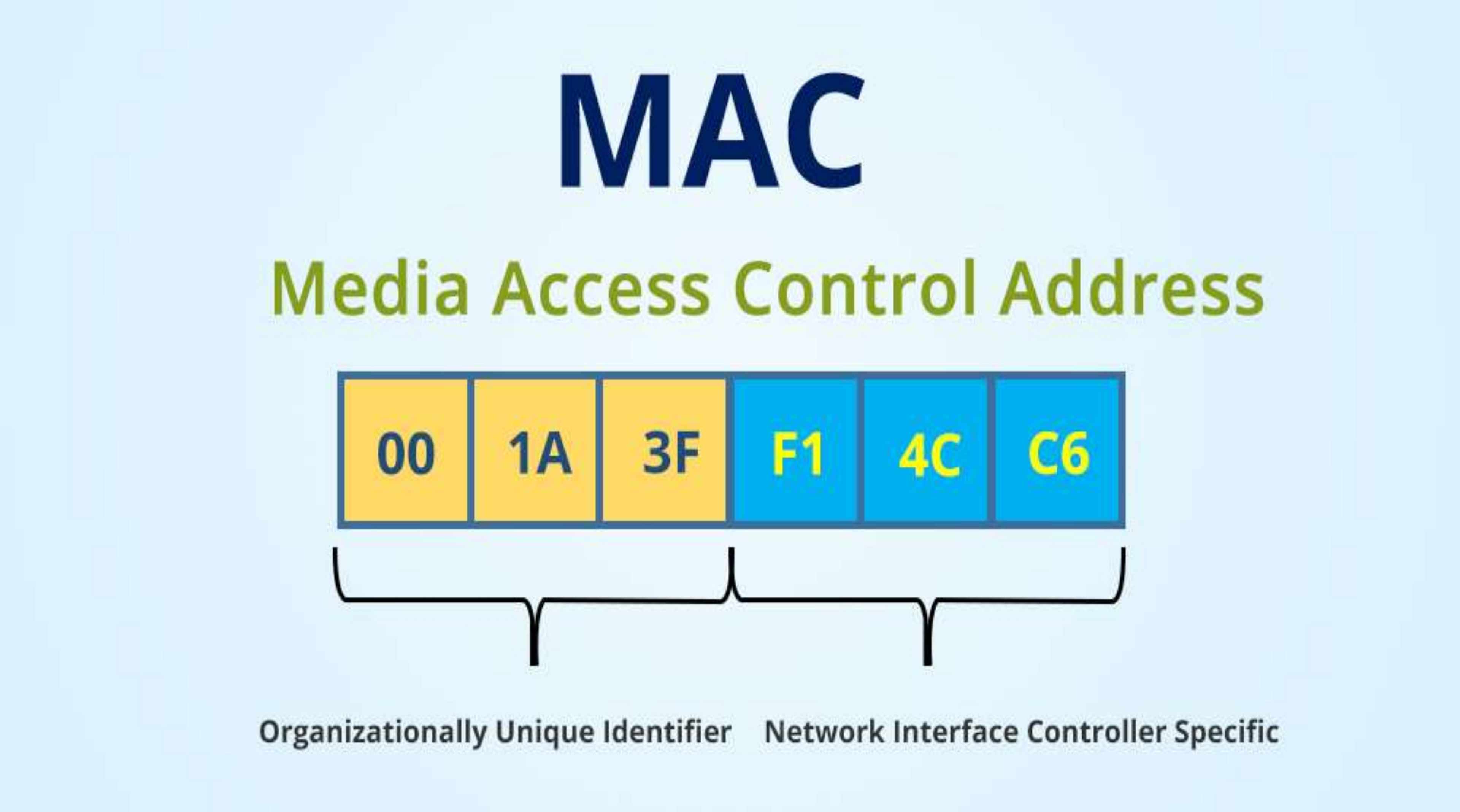
ARP, RARP, BOOTP and DHCP

ARP (Address Resolution Protocol)

ARP (Address Resolution Protocol)

- **Address Resolution Protocol (ARP)** is a crucial protocol in the **Network Layer (Layer 3)** of the **OSI model**
- Maps an **IP address (Logical Address)** to its corresponding **MAC address (Physical Address)** within a local network.

MAC ADDRESS


MAC
Media Access Control Address
00
1A
3F
F1
4C
C6
Organizationally Unique Identifier
Network Interface Controller Specific

Key Points of ARP

- ARP is used in **IPv4 networks**.
- It helps devices in a network to discover the MAC address of another device when only the IP address is known.
- It works in **Broadcast Domain (Local Network)**.
- ARP operates within the **Data Link Layer (Layer 2)** and **Network Layer (Layer 3)**.

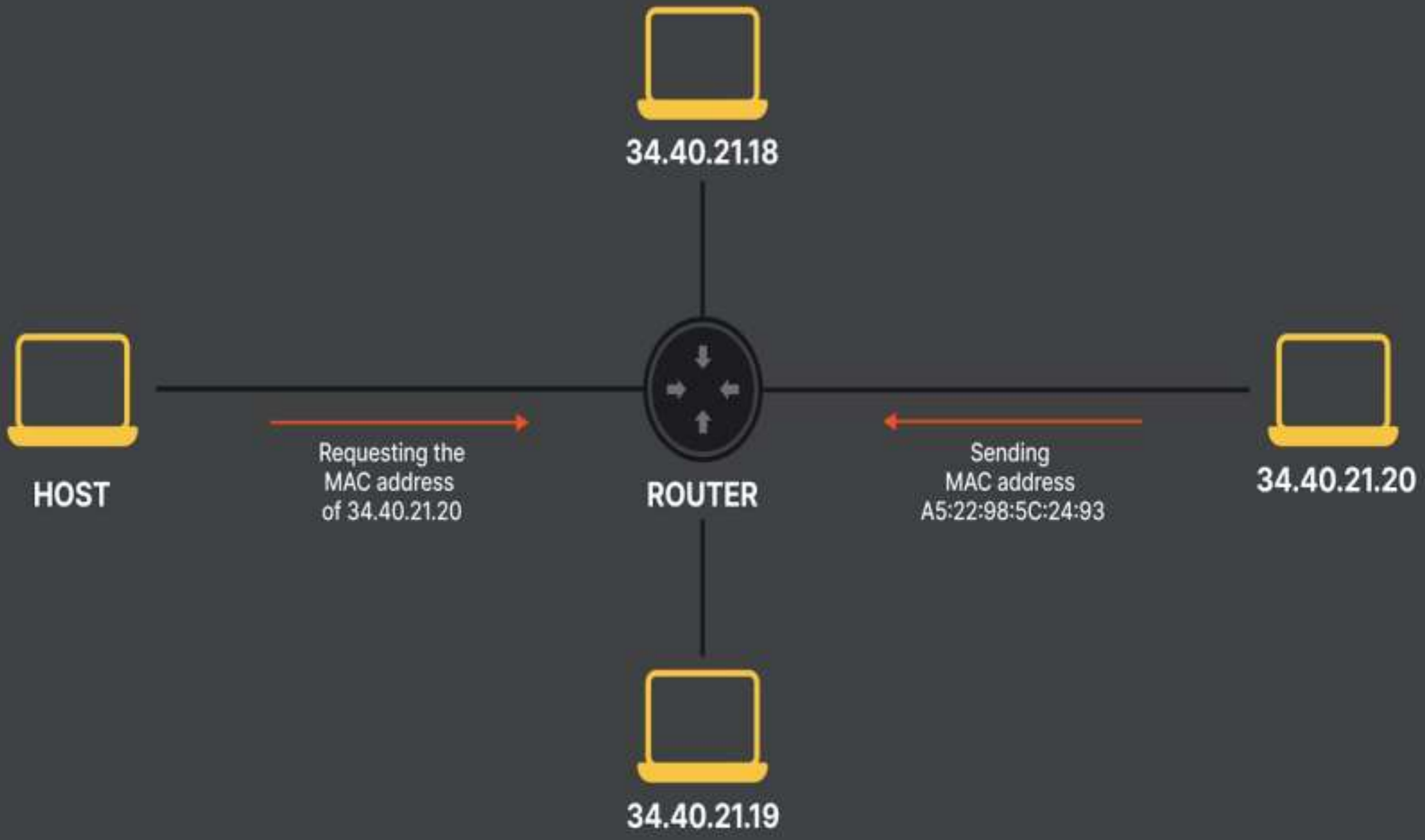
Explanation:

- The sender device wants to communicate with another device on the same network.
- It only knows the **IP address** of the destination but needs the **MAC address** to encapsulate the data in a frame.
- The sender generates an **ARP request packet** that says:
- □ *"Who has this IP address? Tell me your MAC address!"*
- Since the sender doesn't know which device has the requested IP address, the request is broadcasted to **all devices** in the network.

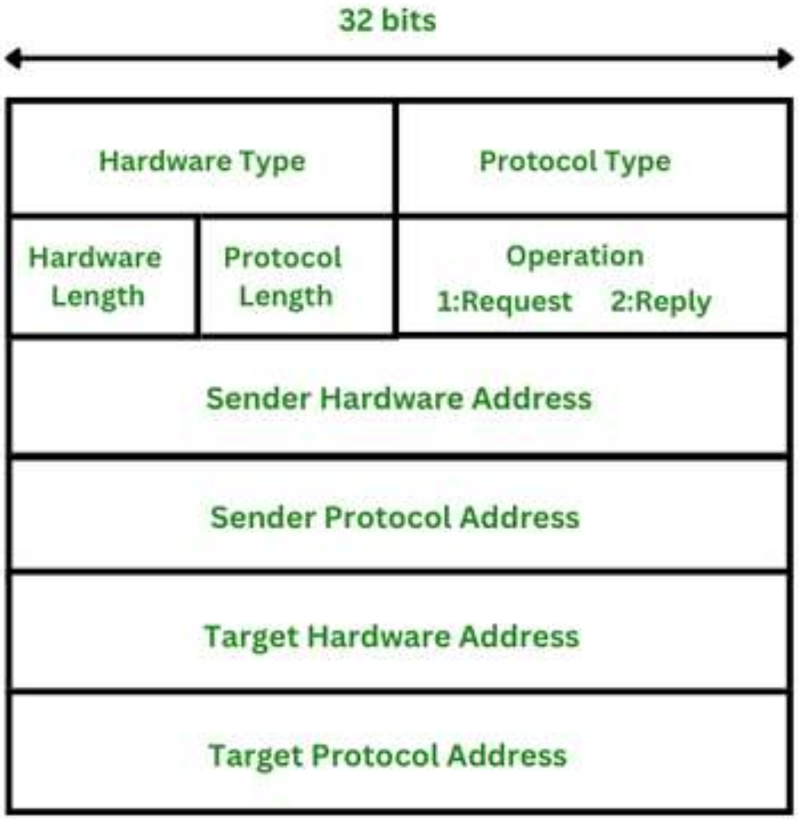
How ARP Works (Step-by-Step Process)

- Request Phase (ARP Request):If a device wants to communicate with another device, it checks its ARP cache.
  - If the MAC address is not found, the device sends a broadcast message to all devices on the network:
  - Example: "Who has IP address 192.168.1.2? Tell me your MAC
  - "This message is sent to FF:FF:FF:FF:FF:FF (Broadcast MAC Address).
- Reply Phase (ARP Reply):The device with the matching IP address responds with its MAC address.
  - Example:"192.168.1.2 is at 08:00:27:AC:13:55“
  - This message is unicasted back to the sender.
- Caching: The sender updates its ARP table (cache) with the IP-to-MAC mapping.
  - The entry is stored temporarily (usually for 20 minutes in Windows).

How Address Resolution Protocol (ARP) Works


34.40.21.18
HOST
Requesting the MAC address of 34.40.21.20
ROUTER
Sending MAC address A5:22:98:5C:24:93
34.40.21.20
34.40.21.19

ARP PACKET FORMAT



32 bits

Hardware Type

Protocol Type

Hardware Length

Protocol Length

Operation
1:Request 2:Reply

Sender Hardware Address

Sender Protocol Address

Target Hardware Address

Target Protocol Address

ARP PACKET

- **Hardware type:** This is 16 bits field defining the type of the network on which ARP is running. Ethernet is given type **1**.
- **Protocol type:** This is 16 bits field defining the protocol. The value of this field for the IPv4 protocol is **0800H**.
- **Hardware length:** This is an 8 bits field defining the length of the physical address in bytes. Ethernet is the value **6**.
- **Protocol length:** This is an 8 bits field defining the length of the logical address in bytes. For the IPv4 protocol, the value is **4**.
- **Operation** (**request or reply):** This is a 16 bits field defining the type of packet. Packet types are **ARP request (1), and ARP reply (2).**
- **Sender hardware address:** This is a variable length field defining the physical address of the sender. For example, for Ethernet, this field is **6 bytes long.**
- **Sender protocol address:** This is also a variable length field defining the logical address of the sender For the IP protocol, this field is **4 bytes long.**
- **Target hardware address:** This is a variable length field defining the physical address of the target. For Ethernet, this field is 6 bytes long. For the ARP request messages, this field is all **Os** because the sender does not know the physical address of the target.
- **Target protocol address:** This is also a variable length field defining the logical address of the target. For the IPv4 protocol, this field is **4 bytes long**.

Commands to Check ARP Table:

- Windows - arp –a
- Linux - arp –n
- MacOS - arp –a
- Cisco Router - show ip arp

Solve

- **Q:** If Host A wants to send data to Host B with IP address **192.168.1.5**, and Host B's MAC address is unknown, how does ARP work?

Answer:

1. Host A sends ARP Request: *"Who has 192.168.1.5? Tell 192.168.1.1"* (Broadcast to FF:FF:FF:FF:FF:FF)
2. Host B replies with: *"192.168.1.5 is at 08:00:27:BC:11:22"* (Unicast to 192.168.1.1)
3. Host A stores the mapping in its ARP table.

RARP

RARP (Reverse Address Resolution Protocol)

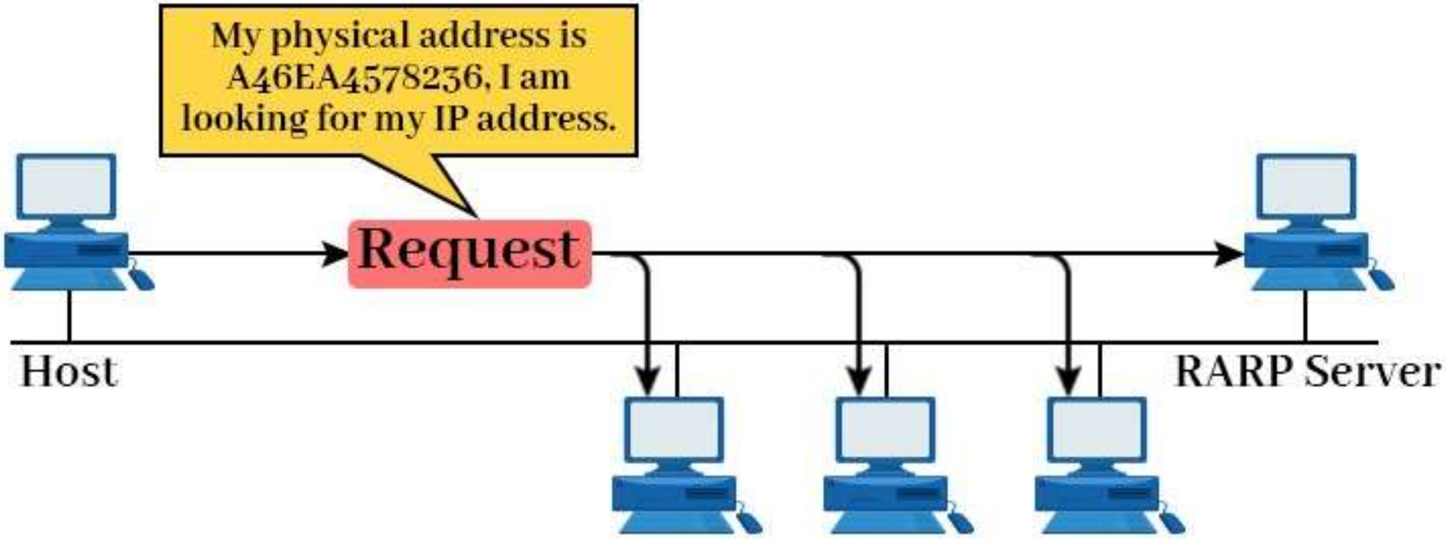
- **Reverse Address Resolution Protocol (RARP)** is a network protocol used to map a **MAC address (Physical Address)** to its corresponding **IP address (Logical Address)**.

Purpose of RARP

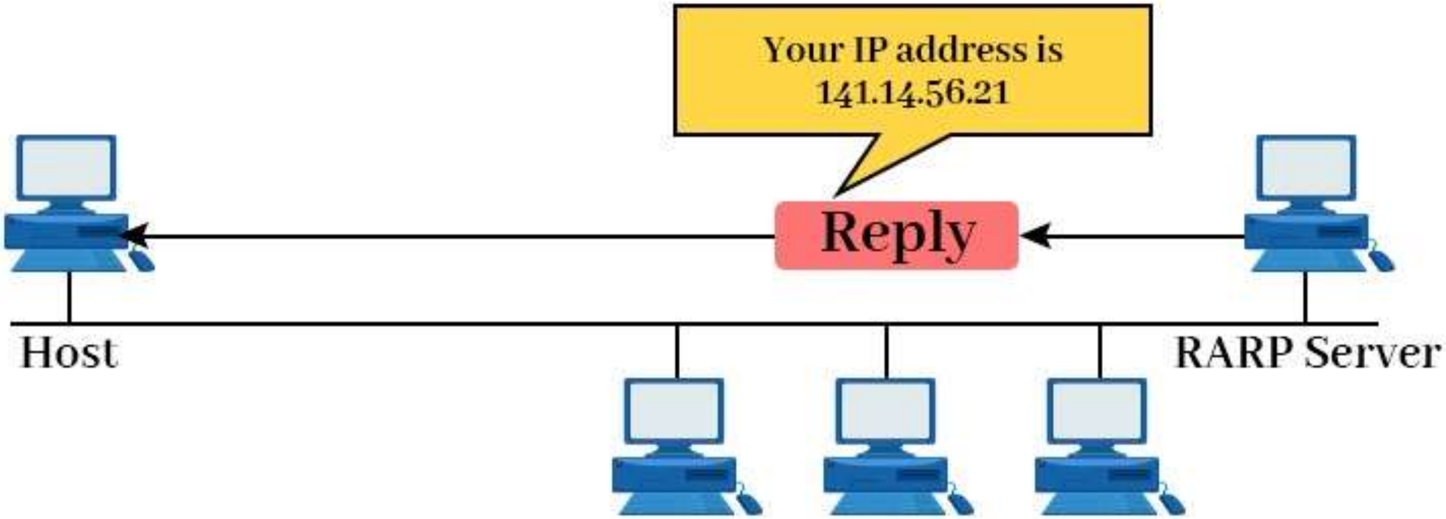
- RARP is used when a device knows its **MAC address** but does not know its **IP address** — typically during the **booting process** of diskless computers or network devices.

How RARP Works (Step-by-Step Process)

- Request Phase (RARP Request):A device (like a diskless workstation) sends a Broadcast RARP request with its MAC address.
  - Example: "Who has IP address for MAC address 08:00:27:AC:13:55?“
- Reply Phase (RARP Reply):The RARP Server (usually a router or specialized server) looks into its database.
  - If the MAC address is found, the server sends the corresponding IP address back to the requesting device.
- Caching: The device stores the obtained IP address temporarily.


My physical address is A46EA4578236. I am looking for my IP address.
Request
Host
RARP Server

RARP Request is Broadcast

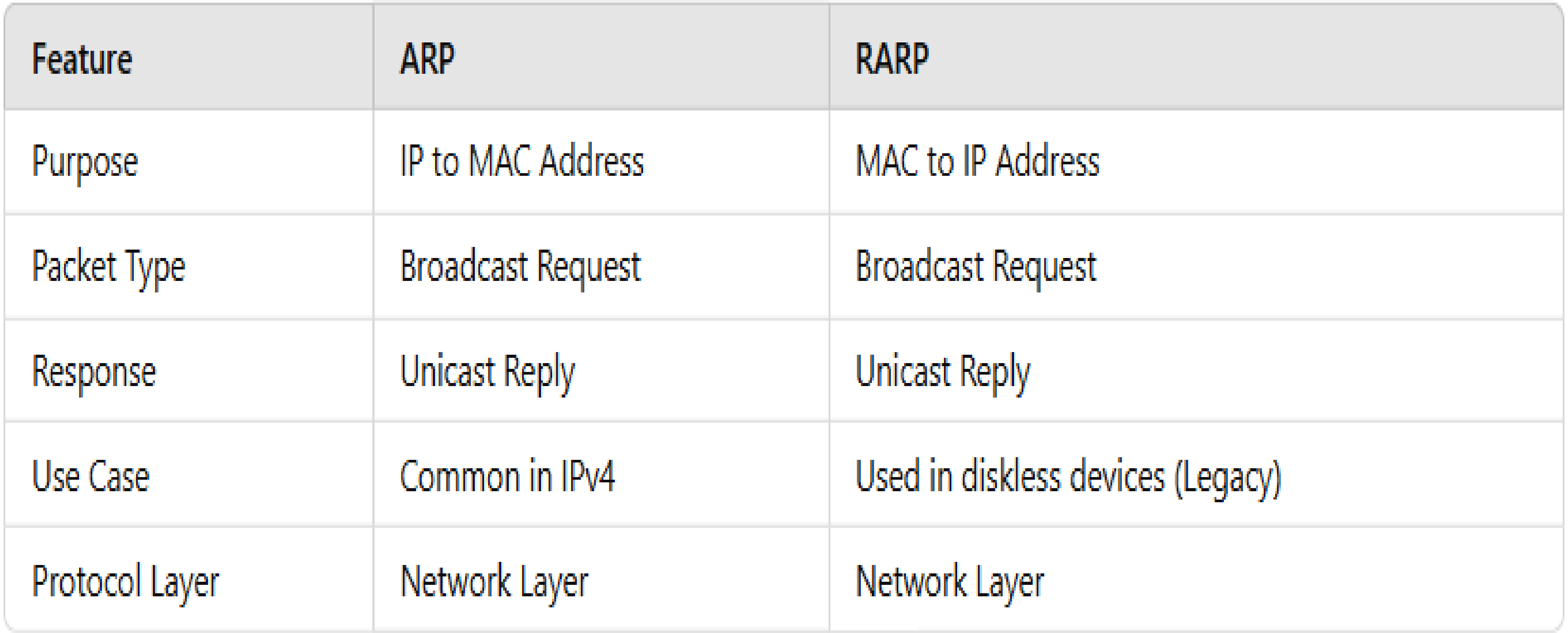


RARP Reply is Unicast

Limitations of RARP

- Only supports **IPv4**.
- Requires a **RARP Server** on every network.
- Cannot handle **Subnetting**.
- Replaced by **BOOTP** and **DHCP** in modern networks.

Differences between ARP and RARP



Feature	ARP	RARP
Purpose	IP to MAC Address	MAC to IP Address
Packet Type	Broadcast Request	Broadcast Request
Response	Unicast Reply	Unicast Reply
Use Case	Common in IPv4	Used in diskless devices (Legacy)
Protocol Layer	Network Layer	Network Layer

BOOTP (Bootstrap Protocol)

BOOTP (Bootstrap Protocol)

- BOOTP (Bootstrap Protocol) is a network protocol used to automatically assign IP addresses, default gateway, and network configurations to diskless computers or network devices during the booting process.

Purpose of BOOTP

- BOOTP was designed to allow **diskless workstations** or **network devices** to automatically obtain:
- IP Address
- Subnet Mask
- Default Gateway
- DNS Server Address
- Operating System Boot File (via TFTP)

How BOOTP Works (Step-by-Step Process)

- BOOTP Request: The client (diskless device) sends a broadcast request on the network asking for its network configuration. The request contains the client's MAC address.
- BOOTP Server Response:
- The BOOTP server checks its database (Configuration File) for the MAC address.
- If the MAC address matches, the server sends a unicast reply with:
  - IP Address
  - Subnet Mask
  - Gateway Address
  - TFTP Server Address
  - Boot File Name
- Boot Process: The client uses the received IP address and downloads the operating system boot file from the TFTP server.

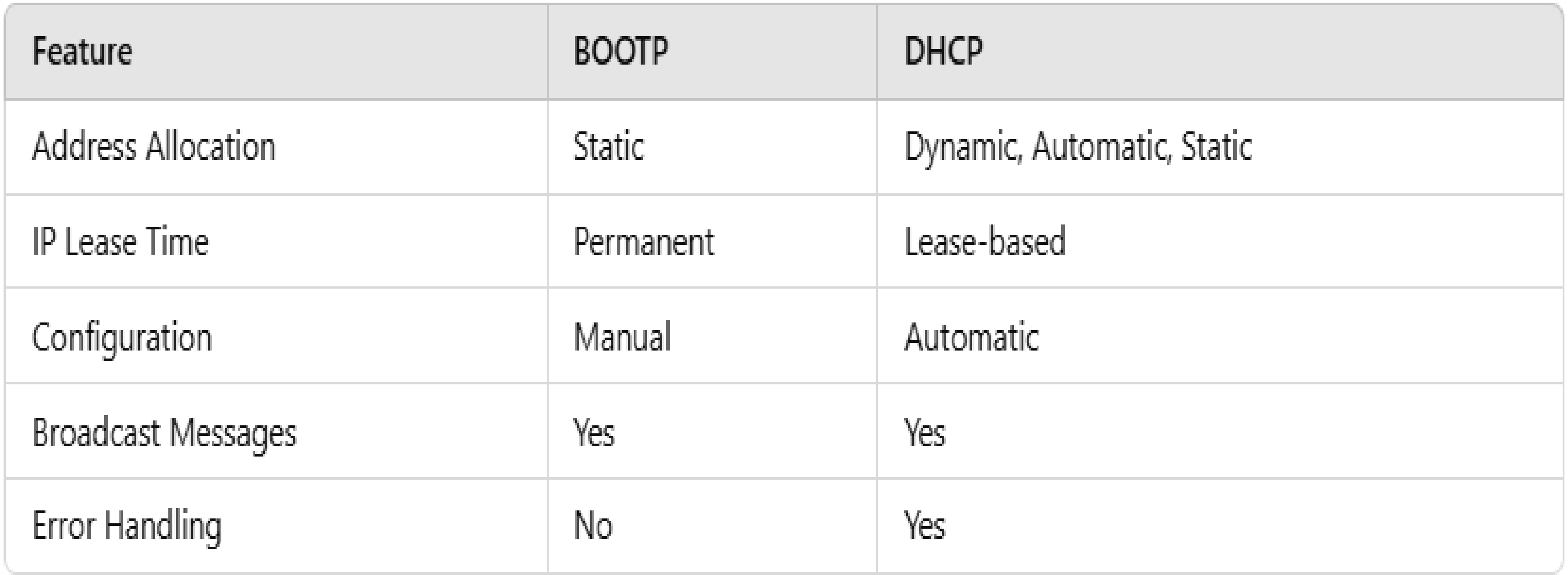
Advantages of BOOTP

- Assigns both **IP address** and **Configuration Information**
- Supports **Static Mapping**
- Better than RARP

Limitations of BOOTP

- Does not support **Dynamic IP Address Allocation**.
- Requires **Manual Configuration** in the BOOTP Server.
- Cannot reuse IP addresses automatically.

Why BOOTP is Replaced by DHCP?



Feature	BOOTP	DHCP
Address Allocation	Static	Dynamic, Automatic, Static
IP Lease Time	Permanent	Lease-based
Configuration	Manual	Automatic
Broadcast Messages	Yes	Yes
Error Handling	No	Yes

DHCP

(Dynamic Host Configuration Protocol)

DHCP

- **DHCP (Dynamic Host Configuration Protocol)** is a network protocol used to automatically assign **IP addresses** and other **network configuration parameters** to devices on a network.
- The primary purpose of DHCP is to **dynamically assign IP addresses** to hosts and reduce the manual effort of network administrators.

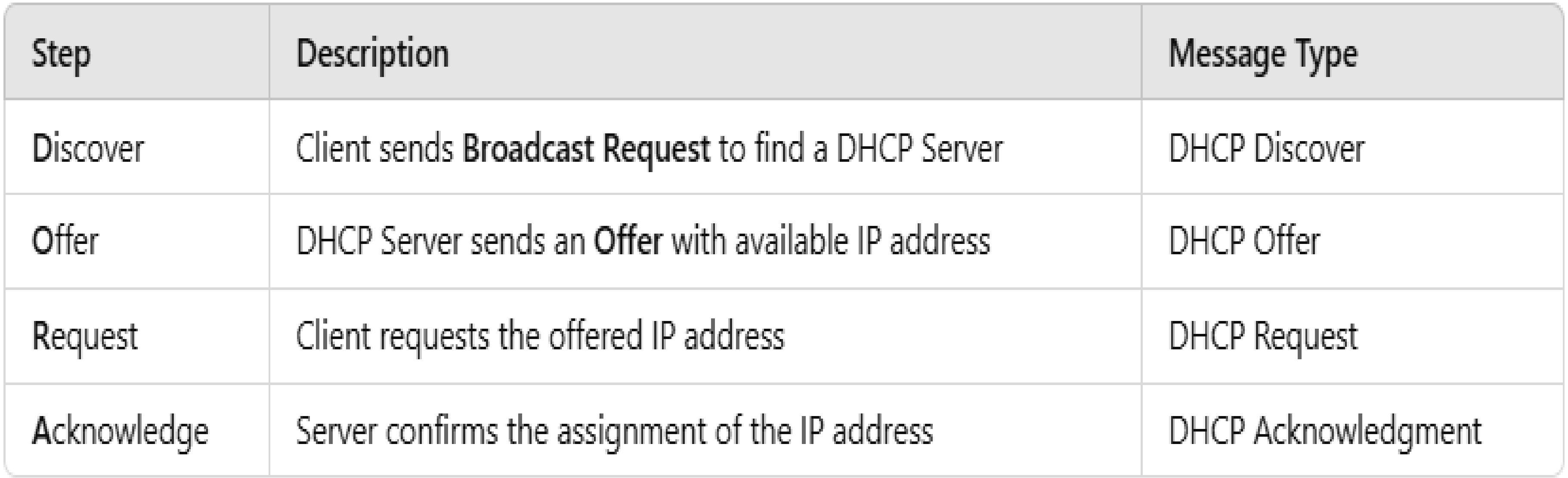
Why DHCP?

- Without DHCP, network administrators would have to manually assign:
  - IP Address
  - Subnet Mask
  - Default Gateway
  - DNS Server
  - Lease Time

How DHCP Works (Step-by-Step Process)

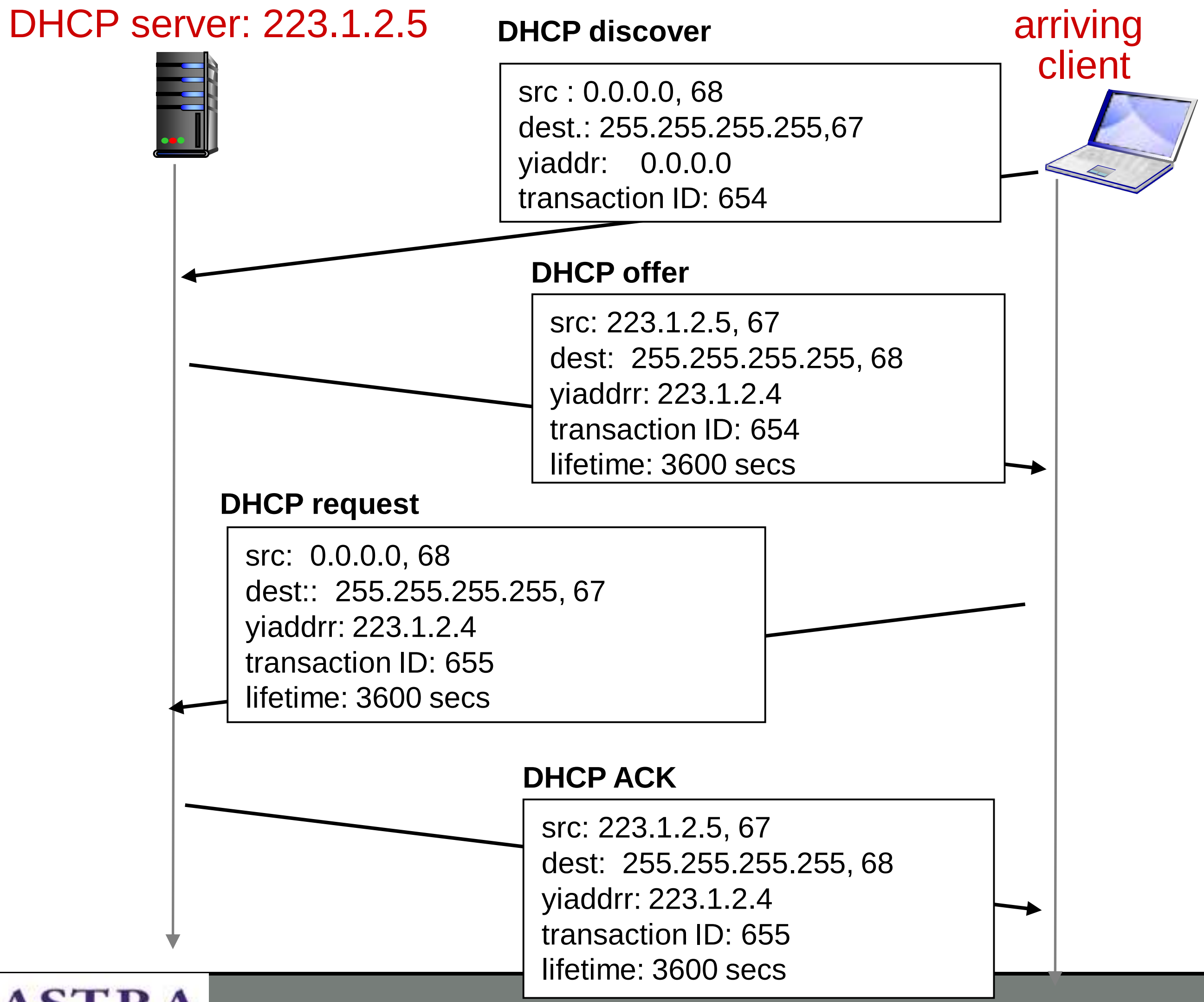
- DHCP uses the DORA Process
- Discover, Offer, Request, Acknowledge

DORA Process in DHCP



Step	Description	Message Type
Discover	Client sends **Broadcast Request** to find a DHCP Server	DHCP Discover
Offer	DHCP Server sends an **Offer** with available IP address	DHCP Offer
Request	Client requests the offered IP address	DHCP Request
Acknowledge	Server confirms the assignment of the IP address	DHCP Acknowledgment

DHCP client-server scenario



DHCP server: 223.1.2.5

arriving client

DHCP discover

src : 0.0.0.0, 68
dest.: 255.255.255.255,67
yiaddr: 0.0.0.0
transaction ID: 654

DHCP offer

src: 223.1.2.5, 67
dest: 255.255.255.255, 68
yiaddrr: 223.1.2.4
transaction ID: 654
lifetime: 3600 secs

DHCP request

src: 0.0.0.0, 68
dest:: 255.255.255.255, 67
yiaddrr: 223.1.2.4
transaction ID: 655
lifetime: 3600 secs

DHCP ACK

src: 223.1.2.5, 67
dest: 255.255.255.255, 68
yiaddrr: 223.1.2.4
transaction ID: 655
lifetime: 3600 secs

Port Numbers Used by DHCP

DHCP Client	**68**
DHCP Server	**67**

Advantages of DHCP

- Automatic IP Address Assignment
- Centralized Management
- IP Address Reuse
- Reduces Configuration Errors
- Supports both **IPv4** and **IPv6**

Limitations of DHCP

- Requires DHCP Server
- Not suitable for small networks
- Security vulnerabilities (IP Spoofing)

IP addresses: how to get one?

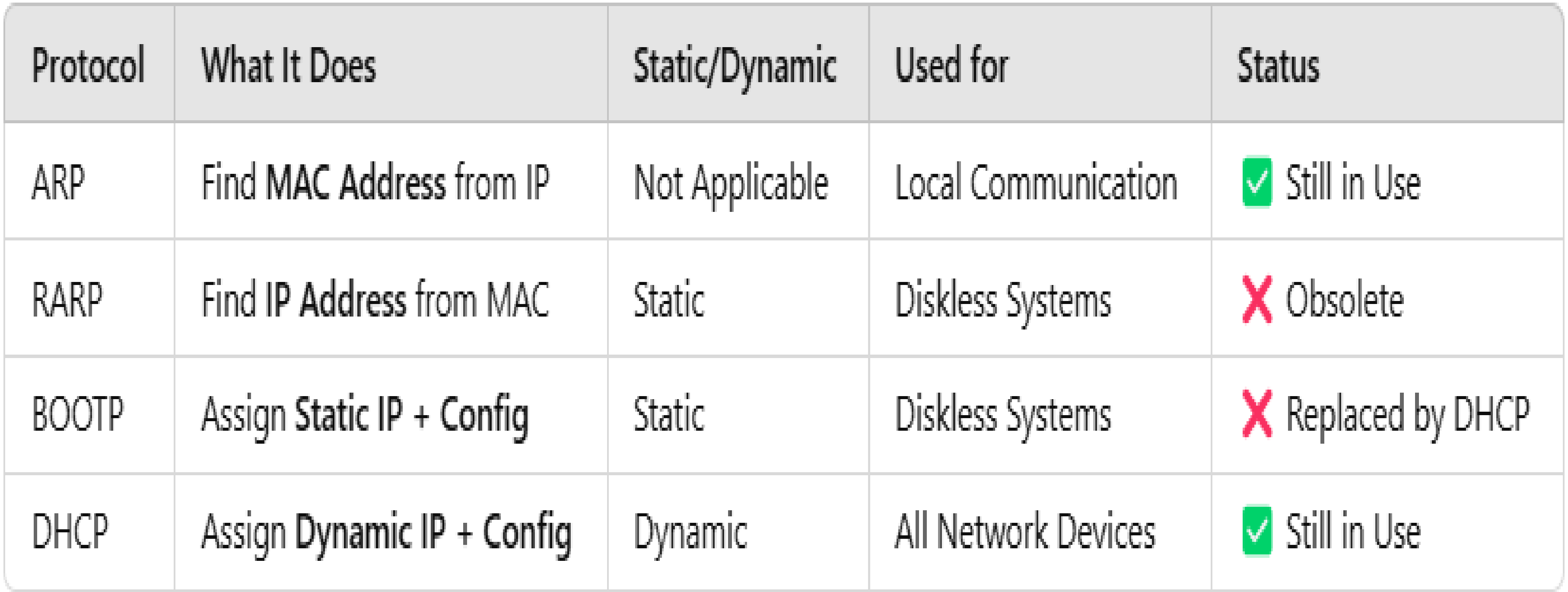
Q: How does a *host* get IP address?

- hard-coded by system admin in a file
  - Windows: control-panel->network->configuration->tcp/ip->properties
  - UNIX: /etc/rc.config
- DHCP: Dynamic Host Configuration Protocol: dynamically get address from as server
  - “plug-and-play”

Which Protocol is Best?

Network Type	Recommended Protocol
Small Network	**Manual Configuration** or BOOTP
Large Network	DHCP
Diskless Devices	**BOOTP** or **RARP (Old Systems)**
Modern Network	DHCP

Summary

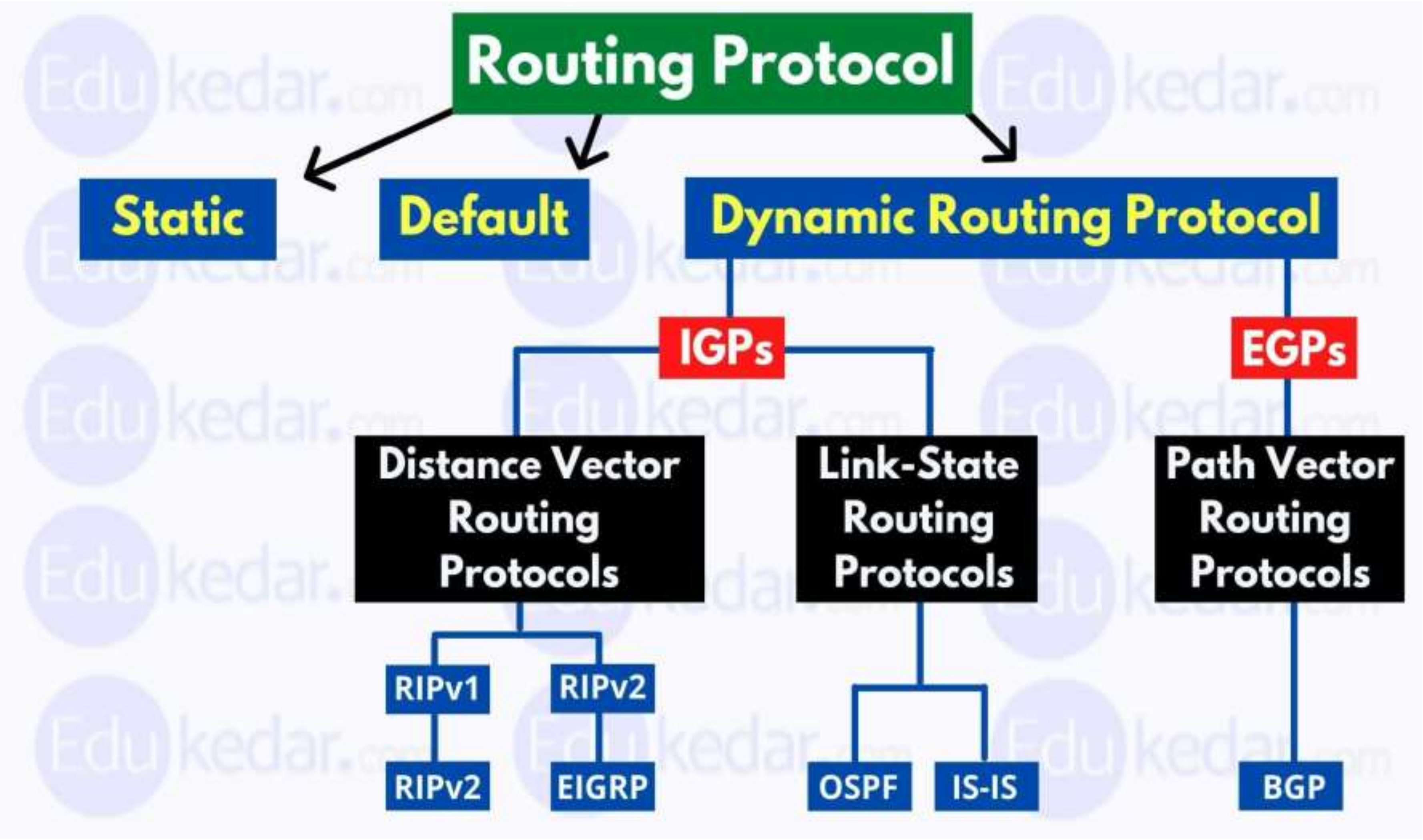


Protocol	What It Does	Static/Dynamic	Used for	Status
ARP	Find **MAC Address** from IP	Not Applicable	Local Communication	✅ Still in Use
RARP	Find **IP Address** from MAC	Static	Diskless Systems	❌ Obsolete
BOOTP	Assign **Static IP + Config**	Static	Diskless Systems	❌ Replaced by DHCP
DHCP	Assign **Dynamic IP + Config**	Dynamic	All Network Devices	✅ Still in Use

Unicast Routing Protocols

- Unicast routing protocols are responsible for determining the best path for data transmission from a single sender (source) to a single receiver (destination) in a network.
- These protocols ensure efficient packet delivery by considering network topology, congestion, and link costs.

Types of Unicast Routing Protocols



Source :https://www.brainkart.com/article/Unicast-Routing-Protocols_13481/

Unicast routing protocols

- Static Routing:
  - Manually configured by network administrators.
  - Suitable for small networks with fixed routes.
  - Does not adapt to network changes dynamically.
- Dynamic Routing:
  - Automatically learns and updates routes based on network conditions.
  - More scalable and adaptable to topology changes.
  - Uses routing algorithms to determine the best path dynamically.

Classification of Dynamic Unicast Routing Protocols

1. Distance Vector Routing Protocols

- Uses hop count as a metric to determine the best path.
- Routers periodically exchange routing tables with their neighbors.
- Example:
  - Routing Information Protocol (RIP) – Uses hop count (max 15) and updates every 30 seconds.

Contd..

2. Link State Routing Protocols

- Uses complete network topology knowledge for route calculation.
- Routers exchange link-state advertisements (LSAs) rather than full routing tables.
- Examples:
  - **Open Shortest Path First (OSPF)** – Uses Dijkstra’s algorithm for shortest path calculation.
  - **Intermediate System to Intermediate System (IS-IS)** – Similar to OSPF, used in ISPs and large networks.

Contd..

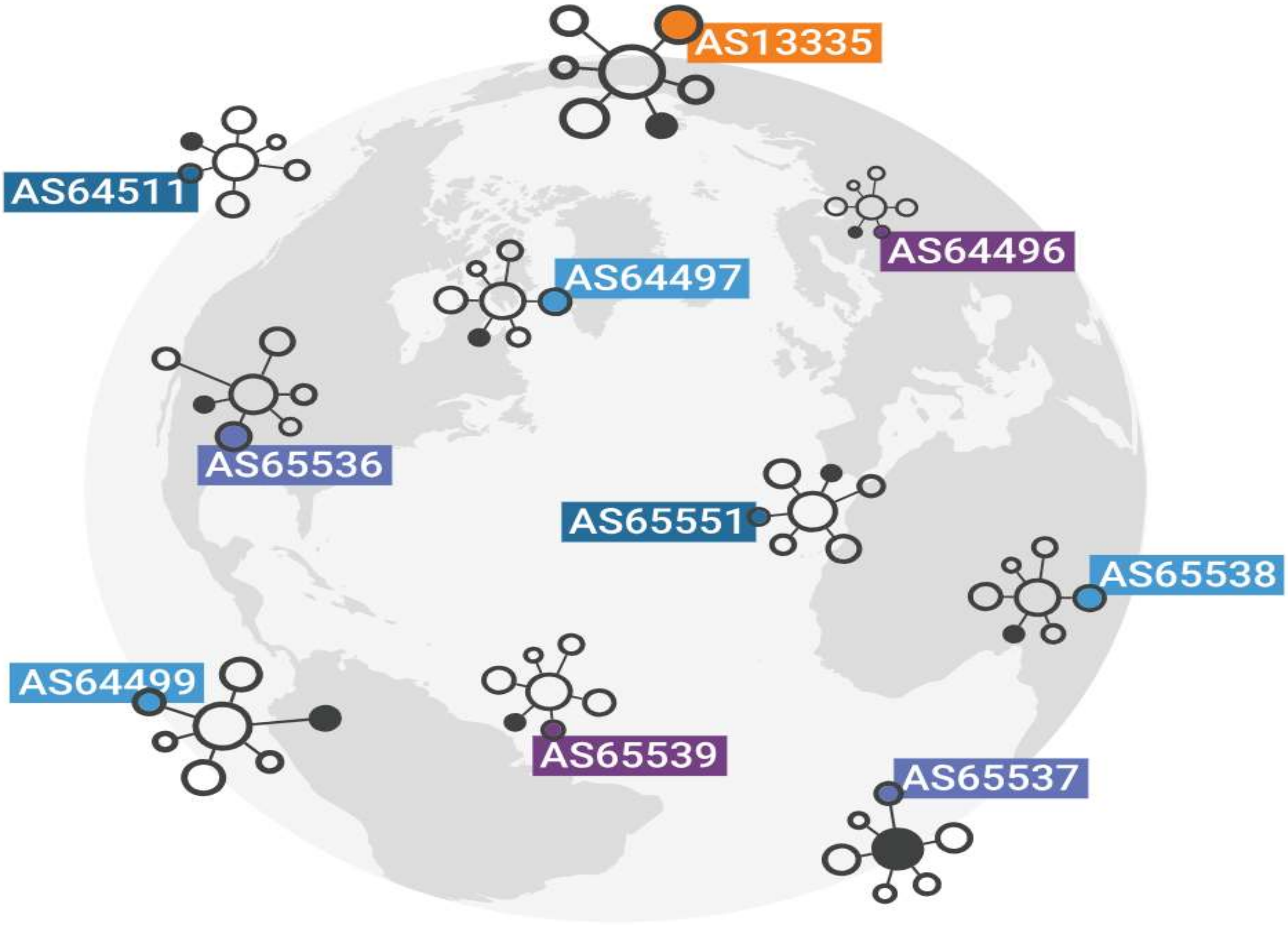
3. Path Vector Routing Protocols

- Used in inter-domain routing where loop prevention is necessary.
- Routers maintain path information rather than just hop counts.
- Example:
  - **Border Gateway Protocol (BGP)** – Used for routing between Autonomous Systems (AS) on the internet.

Autonomous Systems (AS) in Networking

- An **Autonomous System (AS)** is a large network or group of networks under a single administrative control that follows a common routing policy.
- Each AS is identified by a unique **Autonomous System Number (ASN)** assigned by the **Internet Assigned Numbers Authority (IANA)** and managed by **Regional Internet Registries (RIRs)** like ARIN, RIPE NCC, APNIC, LACNIC, and AFRINIC.

Typically, each AS is operated by a single large organization, such as an Internet service provider (ISP), a large enterprise technology company, a university, or a government agency.



Comparison of Unicast Routing Protocols

Protocol	Type	Algorithm	Metric	Convergence Speed	Scalability
RIP	Distance Vector	Bellman-Ford	Hop Count	Slow	Low
OSPF	Link State	Dijkstra’s SPF	Cost (Bandwidth)	Fast	High
EIGRP	Hybrid	DUAL	Composite Metric	Faster than OSPF	Medium
BGP	Path Vector	Path Selection	AS Path, Policies	Slow	Very High