

S.Y.B.Sc.IT SEM III

COMPUTER  
NETWORKS  
(PUSIT303)

BY,

NIKITA MADWAL

# UNIT III

## 5. Network Layer

### 6. Routing



# 5. Network Layer

# Introduction

- The network layer is the third layer in the TCP/IP reference suite model.
- It is responsible for **host-to-host delivery of packets**.
- It provides service to its higher layer i.e., transport layer and receives services from its lower layer i.e., data link layer

# Network Layer Services

## ❖ Packetizing

- The important service is to **create packets from datagram received by the transport layer** at the source and decapsulate the packet at the destination.
- The source host receives the payload from transport layer and adds a header that contains the source and destination address and extra information that is required at the network layer and forwards the packet to the data link layer.
- The **destination host receives the network layer packet from its datalink layer, decapsulates the packet and forwards the payload to the transport layer.**
- The routers can only fragment the packet and not allowed to change any information and add fragmentation information to the packet header.

## ❖ Routing

- This service defines that network layer needs to **find the best route to deliver the packet from source to destination.**
- So, **several algorithms and strategies are used by the network layer to find out the best route.**

## ❖ Forwarding

- Forwarding is an **activity individual router takes when the packet arrives on one of its interfaces.**
- Routing decides about the entire path from source to destination but **forwarding is from one router to another.**
- **A routing or a forwarding table is maintained at each router** for smooth forwarding of the packets.
- When a router receives a packet from one of its attached networks, it needs to forward the packet to another attached network (in unicast routing) or to some attached networks (in multicast routing).
- The forwarding can be done on basis of destination address available in the packet header or based on labels assigned

## ❖ Other Services

### ➤ Error Control

- The **network layer does not directly provide error control** and it is **handled by the higher layers**.
- But it **includes checksum field that checks for corruption only in the header but not the entire packet**.
- However, the ICMP protocol which is an auxiliary network layer protocol provides some kind of error control mechanism.



## ➤ Flow Control

- Flow control **regulates the amount of data a source can send without overwhelming the receiver.**
- If the upper layer at the source computer produces data faster than the upper layer at the destination computer can consume it, the receiver will be overwhelmed with data.
- To control the flow of data, the receiver needs to send some feedback to the sender to inform the latter that it is overwhelmed with data.
- The network layer in the Internet, however, **does not directly provide any flow control.**
- The datagrams are sent by the sender when they are ready, without any attention to the readiness of the receiver.

## ➤ Congestion Control

- Congestion in the network causes the packet to get flooded at one area and as router cannot manage that it discards the packet.
- So, error control mechanism at higher layers cause retransmission of packet and if situation worsens then it ends up no packet reaching the destination.
- **The network layer applies several congestion policies to handle such situation**

## ➤ Security

- When Internet was designed, security was not a need of the hour.
- But as data communication grew to a wider scale, the need for security emerged.
- As network layer was already designed, we **created another virtual layer to implement security called IPSec.**

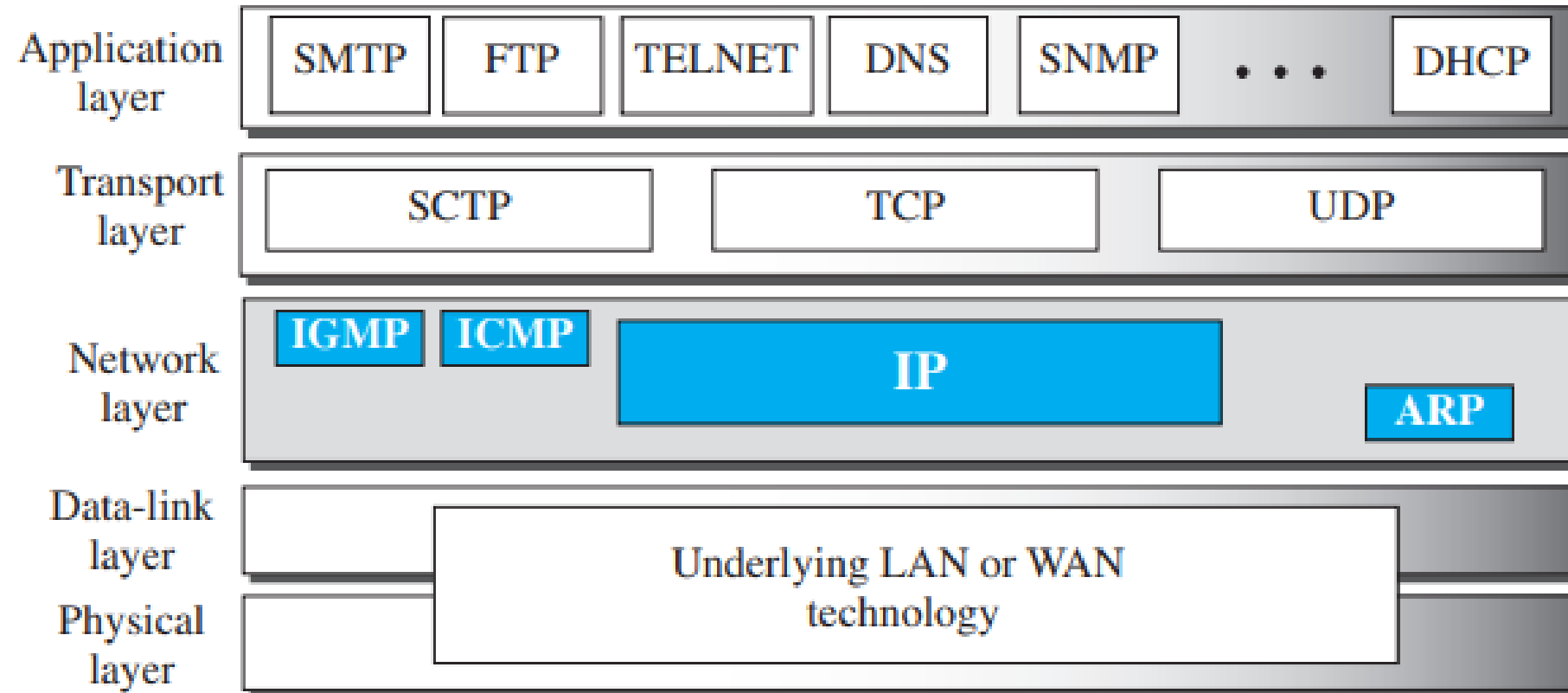
# IPv4 Addresses

- Refer Practical No. 3 PPT

# Internet Protocol

- The network layer has **one main protocol namely Internet Protocol (IP) and three auxiliary protocol** namely Internet Control Message Protocol (ICMP), Internet Group Management Protocol (IGMP) and Address Resolution Protocol (ARP).
- The **IP and ICMP** are available in **version 4 and 6**.
- The IPv4 protocol is **responsible for creation of packet, forwarding, routing and delivering the packet to the destination host**.
- The **ICMPv4** is responsible for **error handling at the network layer**.
- The **IGMP** is responsible for **multicasting in IPv4**.
- The **ARP** is responsible for **IP address to MAC address mapping** and works for the data link layer though it is a network layer protocol.

**Figure 19.1** *Position of IP and other network-layer protocols in TCP/IP protocol suite*

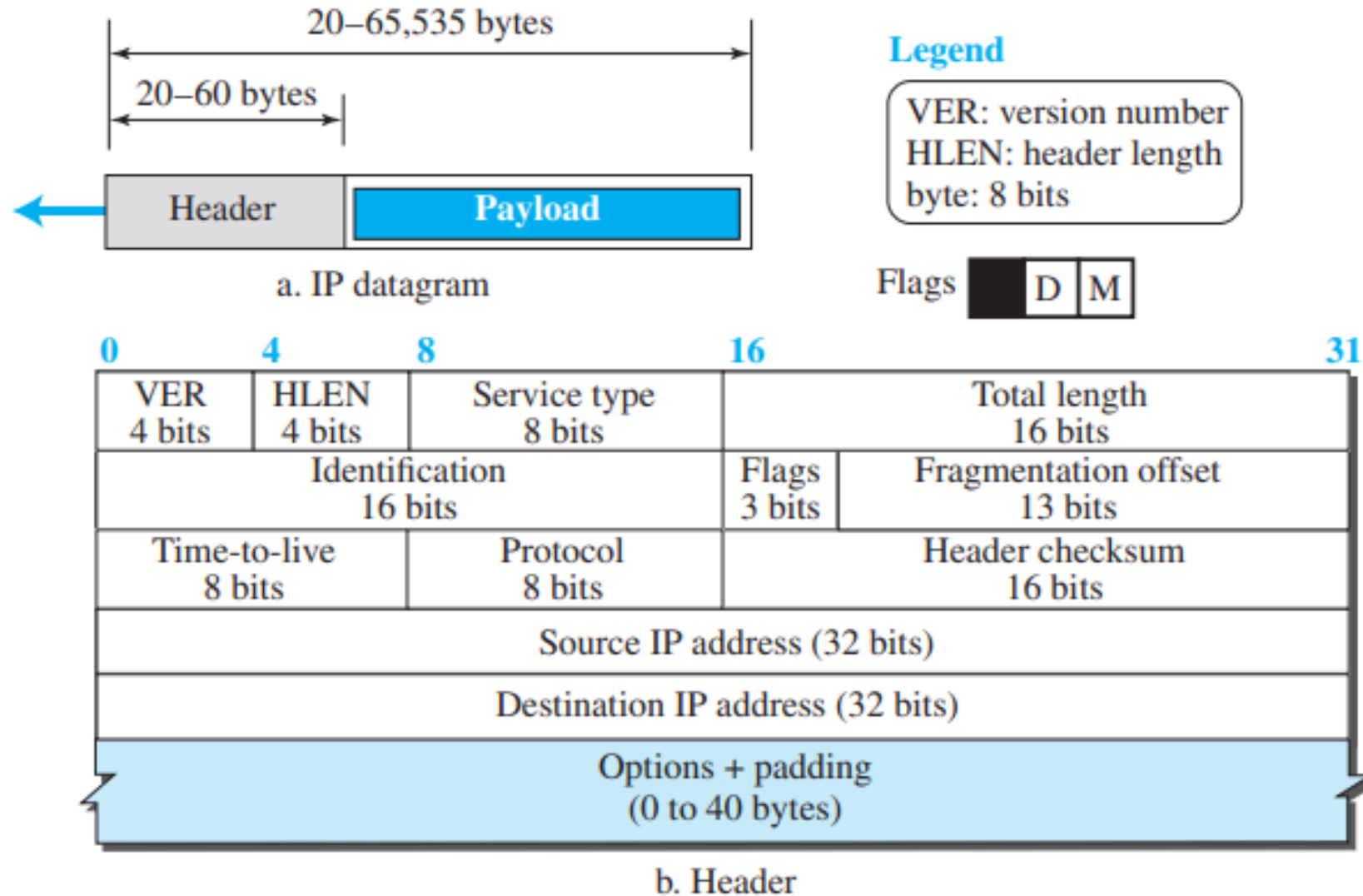


# IPv4 Protocol

- IPv4 is a connectionless, unreliable protocol as each datagram is handled independently and takes a different route to reach the destination.
- The network layer cannot handle large datagrams and so source fragments the datagram into smaller ones.
- The datagrams arrive out of order at destination and reassembly takes place at the destination.
- The intermediate routers can further fragment the datagram but cannot perform reassembly.
- So IPv4 handles creation, forwarding and routing of datagrams.

## ❖ Datagram Format

**Figure 19.2** *IP datagram*



- **IPv4 packets are called datagrams**
- A **datagram** is a variable-length **packet consisting of two parts: header and payload (data)**
- The **header is 20 to 60 bytes in length and contains information essential to routing and delivery**
- **Version Number** : The 4-bit version number (VER) field **defines the version of the IPv4 protocol**, which, obviously, has the value of 4 i.e. version 4.
- **Header Length** : The field HLEN defines the **header length** which is 4-bit field and calculated by multiplying the value of this field by 4.
- **Service Type** : This field is of 8 bits.
- In the original design of the IP header, this field was **referred to as type of service (TOS)**, which defined **how the datagram should be handled**. In the late 1990s, IETF **redefined the field to provide differentiated services (DiffServ)**



- **1. Service Type**
- In this interpretation, the **first 3 bits are called precedence bits**. The **next 4 bits are called type of service (TOS) bits**, and the **last bit is not used**.
- **a. Precedence** is a 3-bit subfield ranging from 0 (000 in binary) to 7 (111 in binary). The **precedence defines the priority of the datagram** in issues such as congestion.
- Such as If a router is congested and needs to discard some datagrams, those datagrams with lowest precedence are discarded first.
- **b. TOS bits** is a 4-bit subfield with each bit having a special meaning.

**Table 20.1** *Types of service*

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

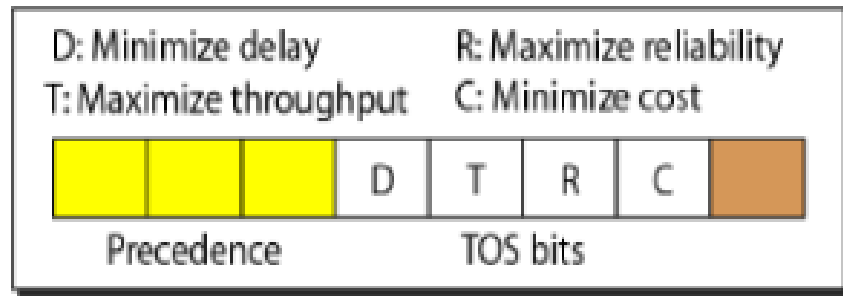
- **2. Differentiated Services**
- In this interpretation, the first **6 bits** make up the **codepoint subfield**, and the last **2 bits** are not used.
- The codepoint subfield can be used in two different ways.
- **a.** When the **3 rightmost bits** are **0s**, the **3 leftmost bits** are interpreted the same as the **precedence** bits in the service type interpretation. In other words, it is compatible with the old interpretation.
- **b.** When the **3 rightmost bits** are not all **0s**, the 6 bits define 64 services based on the priority assignment by the Internet or local authorities according to Table

Category	Codepoint	Assigning Authority
1	XXXXX0	Internet
2	XXXX11	Local
3	XXXX01	Temporary or experiment

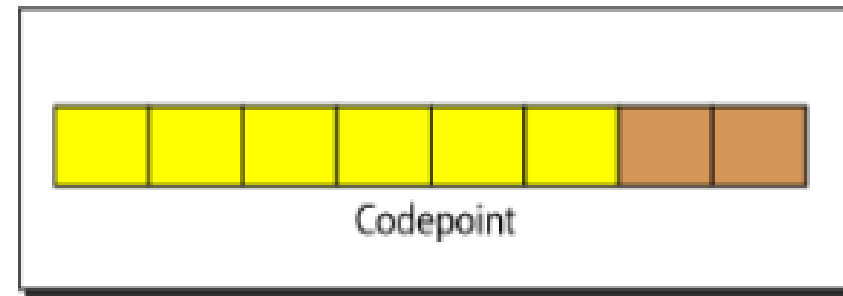
---

**Figure 20.6** *Service type or differentiated services*

---



Service type



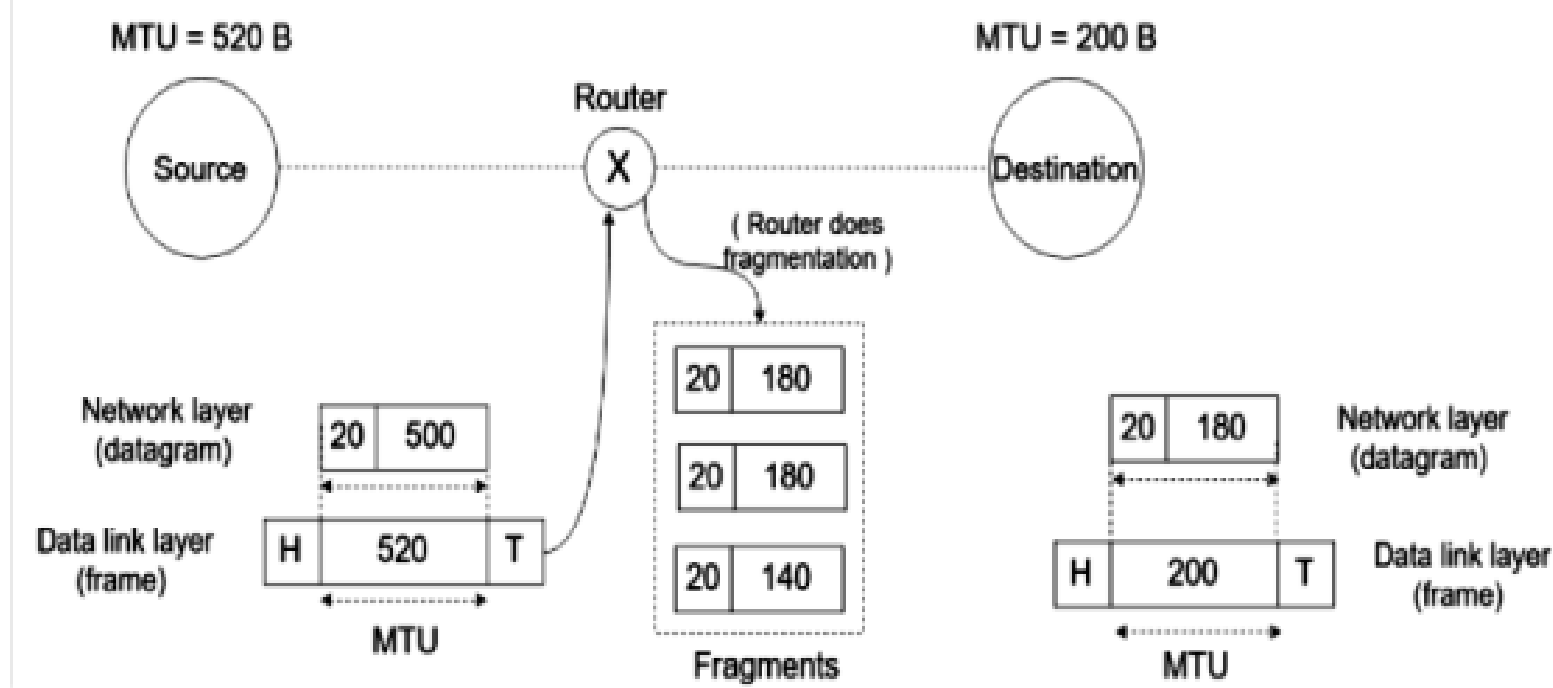
Differentiated services

- **Total length** : This is a 16-bit field that defines the total length (header plus data) of the IPv4 datagram in bytes.
- To find the length of the data coming from the upper layer, subtract the header length from the total length.
- **Identification** : The 16-bit identification field identifies uniquely the datagram created by the source.
- As large message is divided into smaller datagrams, each datagram belongs to the same message needs to be identified.
- **Flags** : This is a 3 bit field of which the first bit is reserved (not used)
- The second bit is called the *do not fragment* bit. If its value is 1, the machine must not fragment the datagram. If its value is 0, the datagram can be fragmented if required
- The third bit is called the *more fragment* bit. If its value is 1, it means the *datagram is not the last fragment* ; there are more fragments after this one. If its value is 0, it means this is the last fragment

- **Fragmentation Offset** : The 13-bit field fragmentation offset defines relative position of the fragment with respect to whole datagram.
- **Time to live** : The time to live field indicates the number of hops i.e., routers visited by the datagram.
  - When a source host sends the datagram, it stores a number which is decremented by 1 by each router that possesses the datagram.
  - If this value, after being decremented, is zero, the router discards the datagram
- **Protocol** : This 8-bit field defines the higher-level protocol that is used in the Transport layer such as TCP, UDP and SCTP, etc.
- **Checksum** : This 8-bit field is used for error detection and to protect the packet (only header) against corruption
- **Source Address** : This 32-bit field defines the IP address of the source.
- **Destination Address** : This 32-bit field defines the IP address of the destination

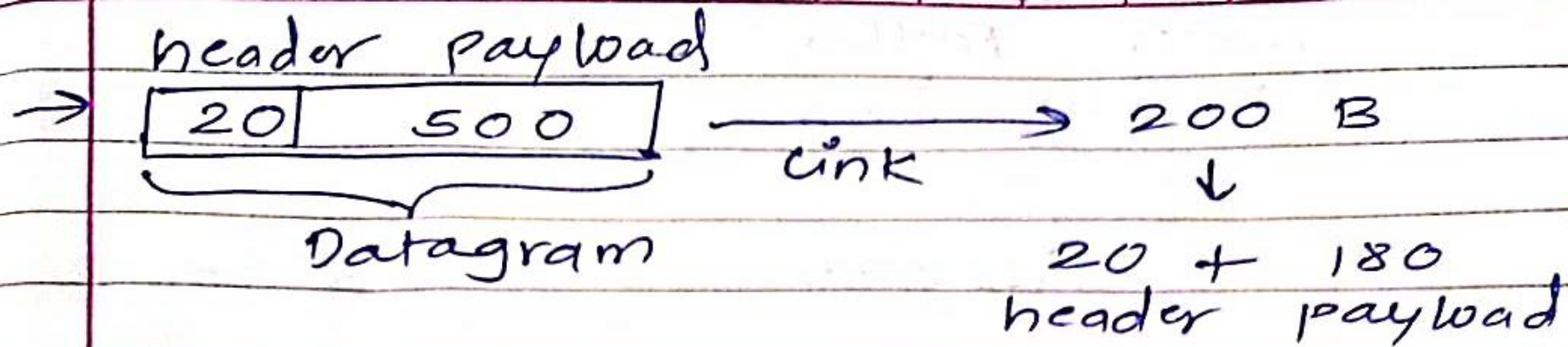
## ❖ Fragmentation

- The network layer **cannot handle large data** and hence the **data is fragmented**.
- The **fragmentation takes place** when the **size of datagram is greater than maximum size of data that can be held a frame i.e., its Maximum Transmission Unit (MTU)**.
- The data flow should not be disrupted and data received from the transport layer is fragmented.
- **Datagram is fragmented by source and intermediate routers and reassembly is taken care by the destination host**.
- **Three fields namely flag, fragmentation offset and total length are changed for the fragmentation process**.



**Figure 12.5– Fragmentation**

- In the figure, the source creates a datagram with header size as 20 bytes and payload with 500 bytes.
- As the datagram reaches the router the, the router fragments into **3 smaller datagrams** with **header size remaining same** for all three fragments and data divided int three sizes namely **180 bytes, 180 bytes and remaining 140 bytes**.



→ 
$$\frac{500}{180} = 2.7 = 3 \text{ fragment}$$

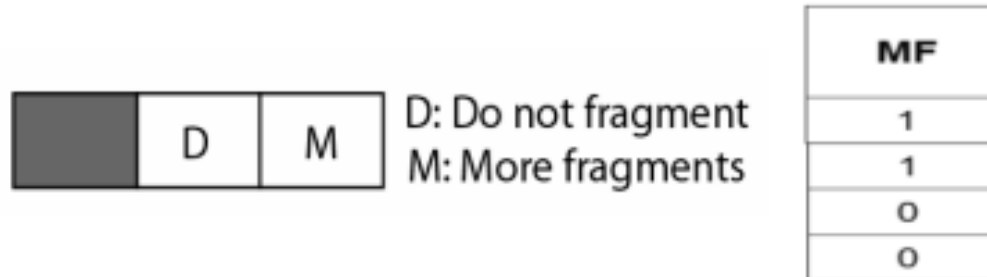
→

$P_3$	$P_2$	$P_1$
140	180	180
20	20	20

→  $[180 + 180 = 360$   
 $\therefore 500 - 360$   
 $= 140]$



- The flag values indicate the fragmentation status



- **D means do not fragment bit.** If **D = 1** the router should not **fragment the datagram** and in that case if router cannot handle such large datagram, then it discards it and sends an ICMP error message to source. If **D = 0** then **datagram can be fragmented if required.**
- **M means more fragment bit.** If **M = 1**, then it is not the last but first or intermediate fragment. If **M = 0** then it indicates it is the **last fragment**

- For the purpose of **reassembly at the destination** host identifies the sequence of datagram from the fragmentation offset
- So, the basic strategy involved in fragmentation is
- The offset field for **first fragment** is **always zero**.
- **Dividing the length of first fragment by 8 gives offset of second fragment.**
- **Dividing the total length of first and second fragment by 8 gives the offset for the third fragment.**
- Continue the same calculation for remaining fragments.

→ fragmentation ~~to~~ offset

P <sub>3</sub>	P <sub>2</sub>	P <sub>1</sub>
140	→ 180	→ 180 → 180/8 = 22.5
20	20	20
<hr/>		
45	22.5	0

└────────→ 180 + 180 = 360/8 = 45

# ARP (Address Resolution Protocol)

- Refer Unit 2 PPT

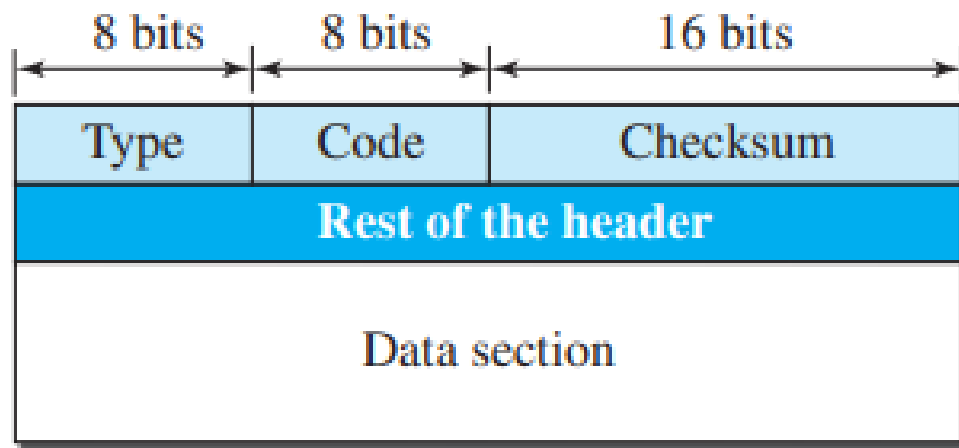
# ICMP

- The IP protocol has no error-reporting or error correcting mechanism. The IP protocol also lacks a mechanism for host and management queries
- The router may discard the datagram because it could not find the route to the destination or TTL field has zero value.
- The Internet Control Message Protocol version 4 (ICMPv4) has been designed to compensate for these deficiencies.
- It is also a network layer protocol but the ICMP message is encapsulated into IP datagram before sending it lower layer.

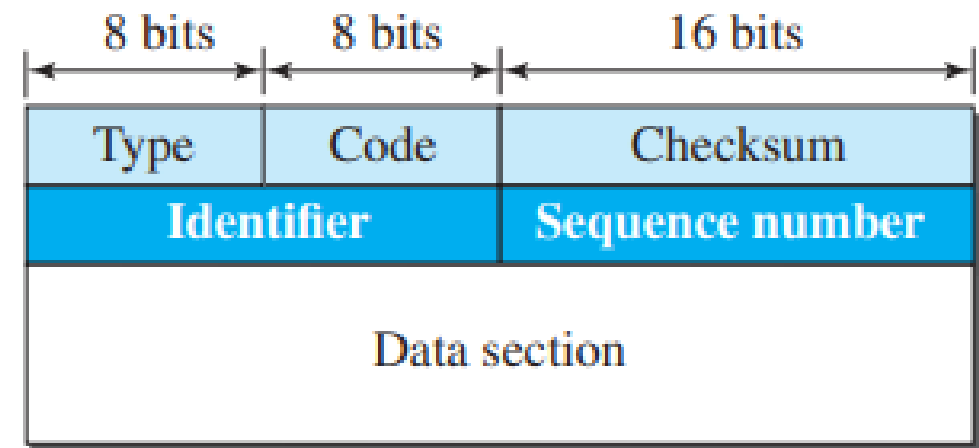
## ❖ ICMP Messages

- ICMP messages are of **two types: error-reporting message and query message**
- The **error-reporting message report the errors and problems** the router and destination face while processing the datagram.
- The **query messages, which occur in pairs, help a host or a network manager get specific information from a router or another host.**
- For example, nodes can discover their neighbors. Also, hosts can discover and learn about routers on their network and routers can help a node redirect its messages.
- **The ICMP message has 8-byte header and variable size data section.**
- **The first 4 bytes of header are common in both types** and next four bytes are different in both.

**Figure 19.8** *General format of ICMP messages*



Error-reporting messages



Query messages

### Type and code values

#### Error-reporting messages

- 03: Destination unreachable (codes 0 to 15)
- 04: Source quench (only code 0)
- 05: Redirection (codes 0 to 3)
- 11: Time exceeded (codes 0 and 1)
- 12: Parameter problem (codes 0 and 1)

#### Query messages

- 08 and 00: Echo request and reply (only code 0)
- 13 and 14: Timestamp request and reply (only code 0)

- The **type** indicates the **type of message** and **code** indicates **reason for message type**.
- The **checksum** is **calculated over header and data** unlike the IPv4 where checksum is calculated for only header content.
- The **data section** in **error-reporting message** carries the **error occurred while processing the datagram** and in query message it contains information depending on the query.



## ❖ Error Reporting Messages

- The **main role of ICMP is to report error** during the processing of datagram as IP is unreliable and incapable of doing so.
- *ICMP cannot correct the errors but only informs about the errors.*
- The higher layers need to take care of the error correction.
- **Errors are reported to source host using the source IP address available in the header.**
- ICMP cannot directly float the error message in the network and so it **forms an error packet and encapsulates it in the IP datagram.**

## ➤The various error reporting message are

### ❑Destination Unreachable

- It is sent by a router when it cannot deliver an IP datagram, the datagram is discarded and the router or the host sends a destination unreachable message back to the source host .

Table 1. ICMP Type 3: Destination Unreachable Codes

Destination Unreachable Code	Description
0	Net is unreachable
1	Host is unreachable
2	Protocol is unreachable
3	Port is unreachable
4	Fragmentation is needed and <b>Don't Fragment</b> was set

## ❑ Source Quench

- It is sent by a destination host or router if it is receiving data too quickly and not able to handle the datagram.
- When a router or a host discards a datagram due to congestion
- The message is a request that the source slow down datagram transmission

## ❑ Redirection

- It is sent by a router to optimize network traffic by redirecting the datagram to another router if it receives a datagram that should have been sent to a different router.

## ❑ Time Exceeded

- It is sent by a router if the datagram has reached the maximum limit of routers through which it can travel.
- This message is generated in 2 cases

- whenever a **router decrements a datagram with a time – to –live value to zero**, it discards the datagram and time exceeded message to original source host
- When the **final destination does not receive all of the fragments in set time**, it discards the received fragments and time exceeded message to original source host

### ❑Parameter Problem

- If a router or the destination host **discovers an ambiguous or missing value in any field of the datagram**, it discards the datagram and sends parameter problem message back to the source

## ❖ Query Messages

- Query messages are independent of IP datagram but again need to be **encapsulated in a datagram as a carrier.**
- Query messages come in pairs and are **used to test the availability and activeness of router in the network.**

❖ The various query messages are

### □ Echo Request & Echo Reply

- It is **used to test destination accessibility and status.**
- A host sends an Echo Request and listens for a corresponding Echo Reply.

### □ Timestamp Request & Timestamp Reply

- It is used to **synchronize the clocks between hosts and to estimate transit time.**

# IPv6

- The IPv4 has been the reigning Internet Protocol version for several decades now even till today.
- But the **address depletion problem of IPv4 has caused IPv6** to come into picture.
- The **IPv4 is running out of room to accommodate all of the unique IP addresses** required for the world's growing number of connected devices.
- The IPv6 is the latest version of the Internet Protocol which identifies devices across the internet so they can be located.
- IPv6 is the next and the **advanced version of the internet protocol used in the network layer**.
- IPv6 provides a larger addressing space.
- The address space reserved in **IPv6 is 128 bits** as compared to only 32 bits in case of IPv4

## ❖IPv6 Addressing

- The **128-bit binary notation** is divided into each **16-bit block** and each block represented by **four hexadecimal digits separated by colon** called the colon hexadecimal notation.

### ➤Notations

- Several notations have been proposed to represent IPv6 addresses when they are handled

#### ❑Dotted Decimal Notation

- Notation is convenient for 4 byte IPv4 addresses. It seems too long for 16 byte
- **221.14.65.11.105.45.170.34.12.234.18.0.14.0.115.255**

#### ❑Colon Hexadecimal Notation

- In this notation, 128 bits are divided into eight sections, each 2 bytes in length. Two bytes in hexadecimal notation require 4 hexadecimal digits.
- **FDEC:BA98:7654:3210:ADBF:BBFF:2922:FFFF**

## ❑ Zero Compression

- It can be applied to colon hex notation **if there are consecutive sections consisting of zeros only**. We can remove all the zeros altogether and replace them with double colon

**FDEC:0:0:0:0:BBFF:0:FFFF** → **FDEC::BBFF:0:FFFF**

## ❑ Mixed Representation

- Sometimes we see a mixed representation of an IPv6 address: **colon hex and dotted decimal notation**
- **FDEC:14AB:2311:BBFE:AAAA:BBBB:130.24.24.18**



## ❖Address Types

- An IPv6 destination address can be unicast, anycast or multicast.

### ❑Unicast address

- It is meant to **configure on one interface** so that you can send and receive IPv6 packets.

### ❑Anycast address

- It is assigned to a **group of interfaces** and a **packet sent to an anycast address is delivered to only one of the nearest hosts.**

### ❑Multicast address

- It is assigned to a **group of interfaces** and the **packet is sent to all interfaces identified by the address.**

## ❖ **Advantages of IPv6 over IPv4**

### ➤ **Larger Address Space**

- It is 128 bit long. hence has a larger address space.

### ➤ **Better header format**

- IPv6 uses a new header format in which options are separated from the base header and inserted, when needed, between the base header and the data.

### ➤ **New options**

- IPv6 has new options to allow for additional functionalities.

### ➤ **Allowance for extension**

- IPv6 is designed to allow the extension of the protocol if required by new technologies or applications.

## ➤ **Support for resource allocation**

- In IPv6, the type-of-service field has been removed, but two new fields, **traffic class and flow label**, have been added to enable the source to request special handling of the packet.

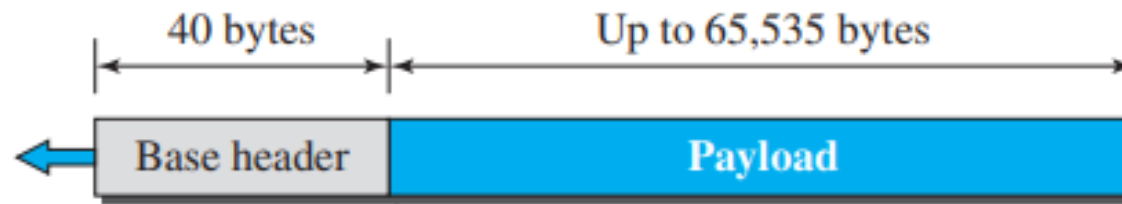
## ➤ **Support for more security**

- The encryption and authentication options in IPv6 provide confidentiality and integrity of the packet.

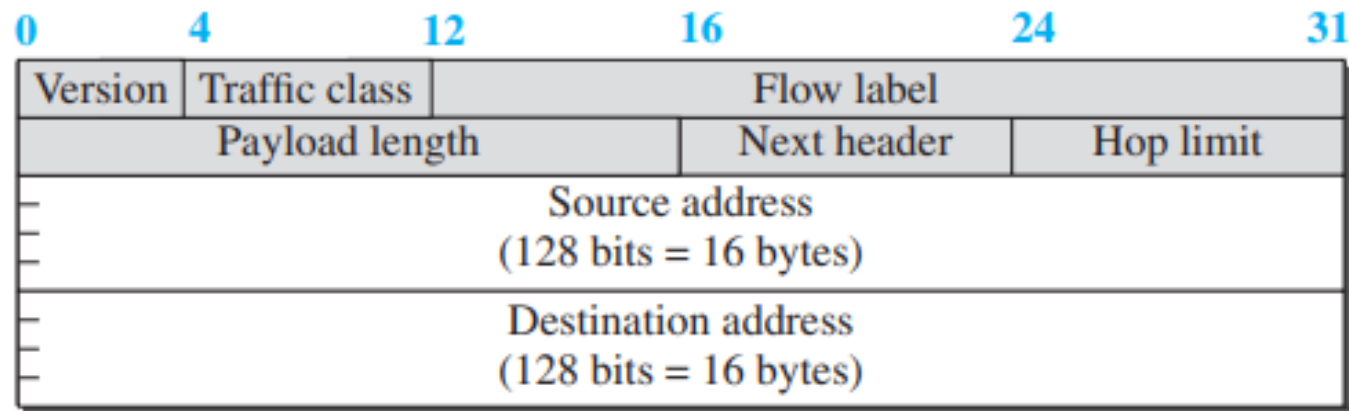
## ❖IPv6 Packet Format

- The IPv6 packet is **composed** of a **base header** followed by the **payload**.
- The **base header occupies 40 bytes**, whereas **payload can be up to 65,535 bytes of information**.

**Figure 22.6** *IPv6 datagram*



a. IPv6 packet



b. Base header

- **Version :** The 4-bit version field defines the **version number** of the IP. For IPv6, the value is 6.
- **Traffic class :** The Traffic Class field indicates **class or priority of IPv6 packet**. It helps routers to handle the traffic based on **priority of the packet**.
- **Flow label :** Flow Label field is used by source to label the **packets belonging to the same flow in order to request special handling** by intermediate IPv6 routers. This makes IPv6 packet to allow IPv6 to work as a connection-oriented protocol.
- **Payload length :** Payload length is 16-bit field that indicates total size of the payload which tells **routers about amount of information a particular packet contains in its payload**.
- **Next extension header :** Next Header indicates type of extension header(if present) immediately following the IPv6 header. This **field is similar to the protocol field in IPv4**

- **Hop limit** : The 8-bit hop limit field serves the **same purpose as the TTL field in IPv4**.
- **Source and destination addresses** : The **source and destination address** are 128-bit field and represents the address of original source and final destination.

## ❖ Extension Header

- The length of the base header is fixed at 40 bytes. However **to give more functionality to the IP datagram**, the base header can be followed by up to six extension headers
- **Hop-by-Hop Option** – It specifies the delivery parameters such as length of datagram, management, debugging and control information at each hop on the path to the destination host.
- **Destination Option** – It specifies packet delivery parameters to the final destination host. Intermediate destination devices are not permitted to access information.
- **Source Routing** – It defines strict source routing and loose source routing for the packet. In loose source routing, the sender specifies only some of the nodes that the packet will pass through, while in *strict source routing*, the *complete set of nodes are determined before sending a packet*.

- **Fragmentation** – As only source host can perform **fragmentation**, it uses the fragment extension header to tell the destination host the size of the packet that was fragmented so that the destination can reassemble the packet.
- **Authentication** – It provides authentication, data integrity, and anti-replay protection. It validates the message sender and ensures the integrity of data.
- **Encrypted Security Payload (ESP)** –It provides data confidentiality, data authentication, and anti-replay protection. It guards against eavesdropping



Order	Header Type	Next Header Code
1	Basic IPv6 Header	-
2	Hop-by-Hop Option	0
3	Destination Option	60
4	Source Routing	43
5	Fragmentation	44
6	Authentication	51
7	Encrypted Security Payload	50

**Table 14.3 – IPv6 Next Header Code**

# IPv4 vs IPv6

## IPv4

- IP address is of 32 bits (4 bytes)
- Size of the header can range from 20-60 bytes depending upon the options
- Fragmentation can be done by sender in between routers
- Fragmentation fields are present inside the header
- Checksum field is present in header
- IPsec support is optional
- Flow label field is not present

## IPv6

- IP address is of 128 bits (16 bytes)
- Size of the header is always fixed i.e. 40 bytes
- Fragmentation can be done by original sender only
- Fragmentation fields are included in options
- Checksum field is present in options
- IPsec support is built-in
- Flow label field speeds up the routing process

# 6. Routing

# Routing

- The network layer is responsible for host-to-host delivery of packets.
- The **unicast routing governs the transmission of packet to only one destination** (one to one) whereas **multicast routing governs the transmission of packet to several destinations** (one to many).
- Unicast routing can be implemented using hierarchical routing where we route in steps

## ❖ Unicast Routing Protocols

- A internet is a combination of networks connected by routers
- When **datagram** goes from a source to a destination, it will **probably pass through many routers until it reaches the router attached to the destination network**
- A router receives a **packet** from one router and passes it to **another router**
- The routers are able to take the decision on the basis of the information that is provided by routing table.
- The **routing table** either can be a static or dynamic
- A static routing table is created and updated manually
- A dynamic routing table is the one that is created once manually but is updated automatically whenever there is some change in the internet

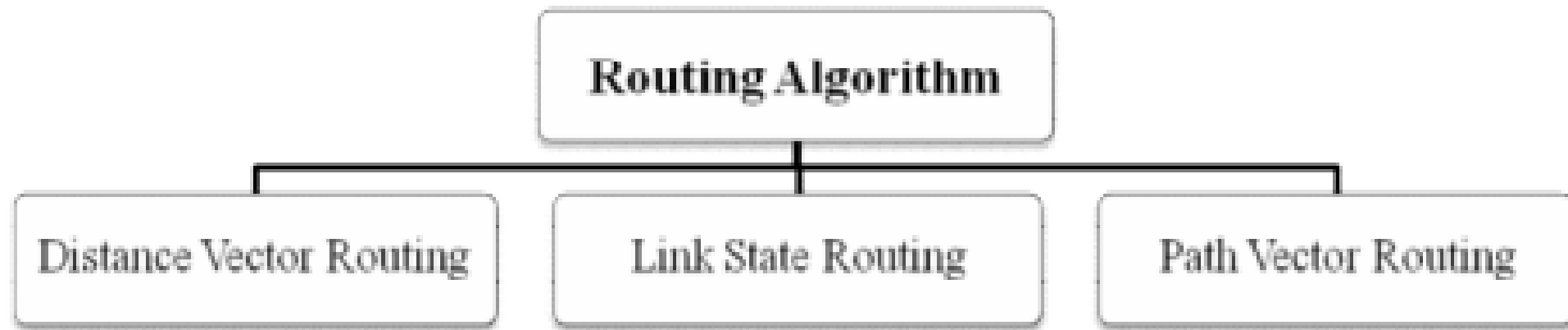
- *Routing protocols have been developed in order to create and update dynamic routing tables*
- *A routing protocol is a combination of rules and regulations using which routers share their information and inform each other of changes*

## ❖ Intra and Inter-domain Routing

- An internet today has become so huge that one routing protocol can not do the job of updating the routing table
- Hence, the internet is divided into smaller parts known as *Autonomous System*.
- An *autonomous system is a group of networks and routers connected to each other*.
- Routing within an autonomous system is known as interior routing or intra domain routing. *RIP and OSPF are intra domain routing protocol*.
- Routing between multiple autonomous systems is known as exterior routing or inter domain routing. BGP are inter-domain routing protocol.

## ❖ Routing Algorithms

- Routing algorithms are meant for determining the routing of packets in a node.
- Several routing algorithms have been devised.



**Figure 13.3 – Types of Routing Algorithms**



## ❖Distance-vector Routing

- The distance vector protocol is the oldest routing protocol in practice. With distance vector routes are advertised based upon the following **characteristics**:
- **Distance** - How far the destination network is based upon a metric such as **hop count**.
- **Vector** - The **direction** (next-hop router or egress interface) required to get to the destination.

## ❖Link-state Routing

- link state routing ,**relies on each node advertising/flooding the state** (i.e. delay, bandwidth etc) of their **links to every node within the link state domain**.
- This results in each node building a complete map of the network (**shortest path tree**),

## ❖ Path Vector Routing

- Path vector (PV) protocols, are used **across domains** aka autonomous systems.
- In a path vector protocol, a router does not just receive the distance vector for a particular destination from its neighbor; instead, **a node receives the distance *as well* as path information**

# RIP (Routing Information Protocol)

- Refer Practical No. 6 PPT

# OSPF (Open Shortest Path First)

- Refer Practical No. 6 PPT

# BGP (Border Gateway Protocol)

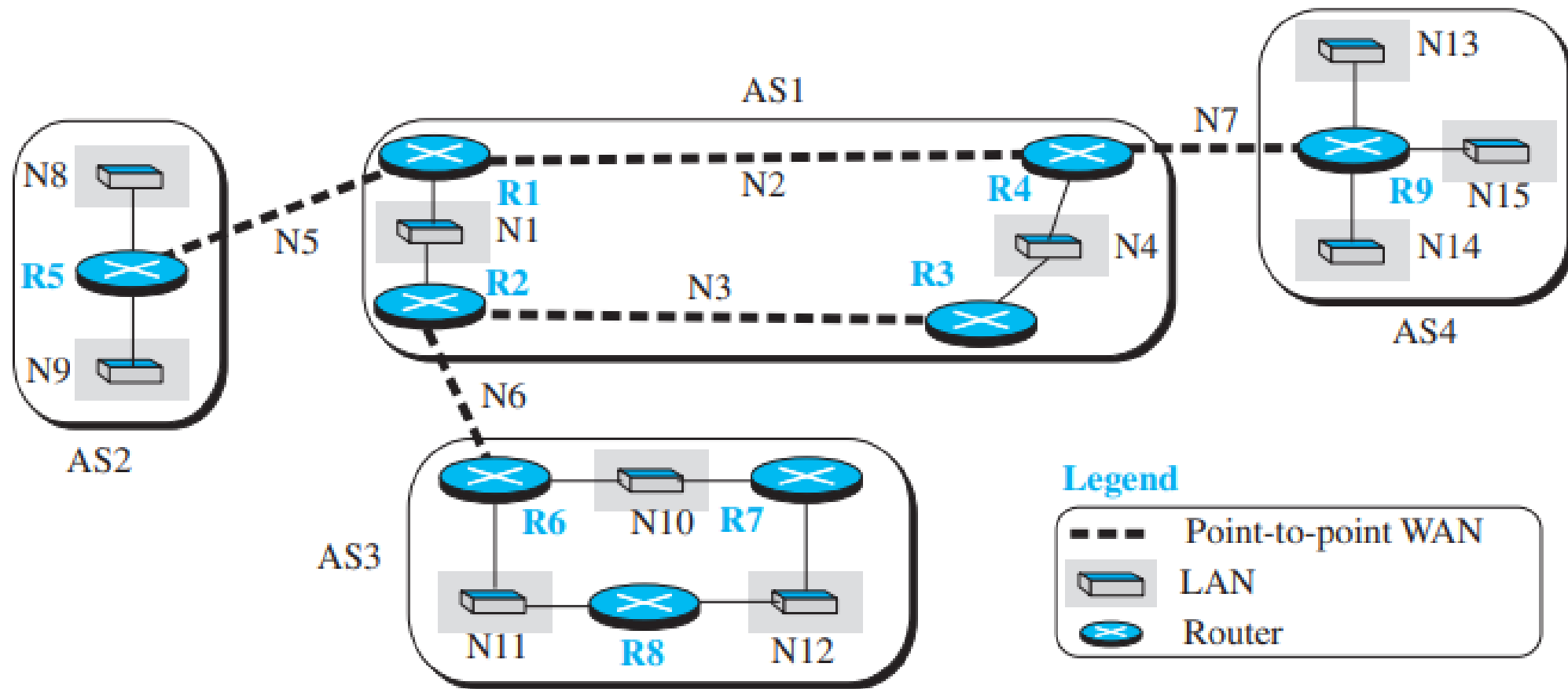
- BGP is an **exterior or inter-domain routing protocol** that **handles the job of routing between autonomous system**
- BGP is based on **path vector routing** i.e. **is uses this algorithm to initially create and update its routing table**

## ❖ Path Vector Routing

- Path vector routing is **exterior routing protocol** proved to be **useful for inter –domain or inter-AS (Autonomous System) routing.**
- In path vector routing, a **router has a list of networks that can be reached with the path (list of ASs to pass) to reach each one**

- The figure shows the 4 autonomous system using path vector routing

**Figure 20.24** *A sample internet with four ASs*



- Consider the above network with AS1 as transient AS and AS2, AS3 and AS4 as stub AS
- Each AS uses intradomain protocol such as RIP or OSPF for routing internally within the AS.
- But to know how *to route packets to network in another AS, we require the BGP protocol.*
- There are *two variants of BGPv4 protocol.*
- The external BGP (eBGP) is run on each border router i.e., the one at the edge of each AS which is connected to a router at another AS.
- Another version is *internal BGP (iBGP) run on all routers.*
- So, *border routers run three protocols namely intradomain, iBGP and eBGP and other routers run two protocols namely intradomain and iBGP.*

- The border *routers that run the eBGP* are called as *BGP peers or speakers*.
- The *speaker node that speaks on behalf of the node*
- The speaker node is *responsible for creating and updating the routing table of the respective autonomous system*