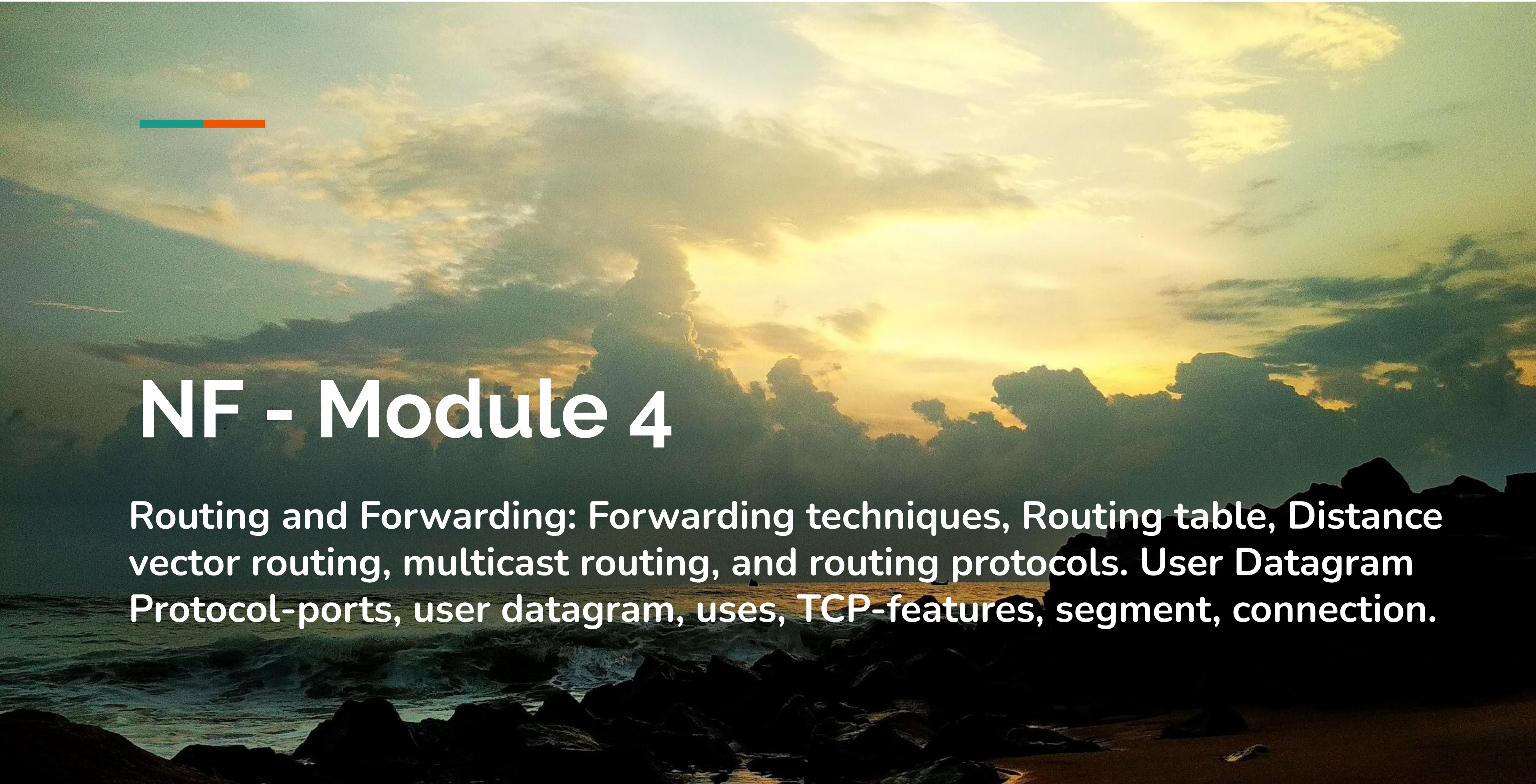


NF - Module 4

Routing and Forwarding: Forwarding techniques, Routing table, Distance vector routing, multicast routing, and routing protocols. User Datagram Protocol-ports, user datagram, uses, TCP-features, segment, connection.

ROUTING & FORWARDING

Network layer is the third layer in the OSI model of computer networks. It’s main function is to transfer packets from the source to the destination. At the source, it accepts a packet from the transport layer, encapsulates it in a datagram/packet and then deliver the packet to the data link layer so that it can further be sent to the receiver. At the destination, the datagram is decapsulated, the payload data is extracted and delivered to the corresponding transport layer.

Features : Main responsibility of Network layer is to carry the data packets from the source to the destination.

1. If the packets are too large for delivery, they are fragmented i.e., broken down into smaller packets. (called packetizing)
2. It decides the route to be taken by the packets to travel from the source to the destination among the multiple routes available in a network (also called as routing).
3. The source and destination addresses are added to the data packets inside the network layer.

Routing and Forwarding –

These are the two services offered by the network layer. In a network, there are a number of routes available from the source to the destination. The network layer must determine the route or path, the packets should take as they flow from sender to receiver. **The network layer specifies some strategies, which find out the best possible route. This process is referred to as Routing**. Routing is performed by a special device known as a router. There are a number of routing protocols which are used in this process and they should be run to help the routers coordinate with each other and help in establishing communication throughout the network.

Forwarding: Forwarding means to place the packet in its route to its destination. Forwarding is simply defined as the action applied by each router when a packet arrives at one of its interfaces. When a router receives a packet from one of its attached network, it needs to be forwarded to another network. Forwarding requires a router to have a routing table. When a router has received a packet to be forwarded, it looks at this table to find the route to the final destination.

However, this simple solution is impossible today in the Internet, because the number of entries needed in the routing table would make table lookups inefficient.

So Forwarding is the process of sending the packet towards the destination based on routing information, while Routing means finding the best path for a packet from the source to its destination.

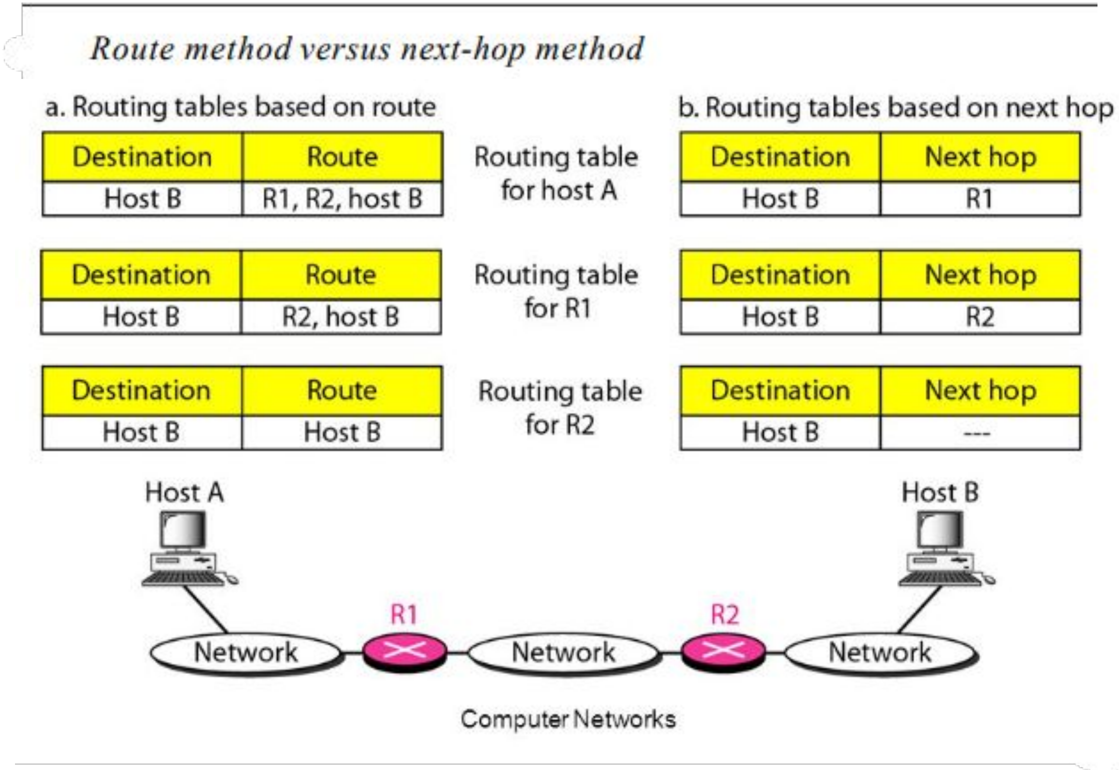
Forwarding Techniques

Several techniques can make the size of the routing table manageable and also handle issues such as security. We briefly discuss these methods here.

- Next-Hop Method
- Network-Specific Method
- Default Method

1. Route Method Versus **Next-Hop Method** :

One technique to reduce the contents of a routing table is called the **next-hop method**. In this technique, the routing table holds only the address of the next hop instead of information about the complete route (as in route method).



Route method versus next-hop method

a. Routing tables based on route

b. Routing tables based on next hop

Routing table for host A

Destination	Route
Host B	R1, R2, host B

Destination	Next hop
Host B	R1

Routing table for R1

Destination	Route
Host B	R2, host B

Destination	Next hop
Host B	R2

Routing table for R2

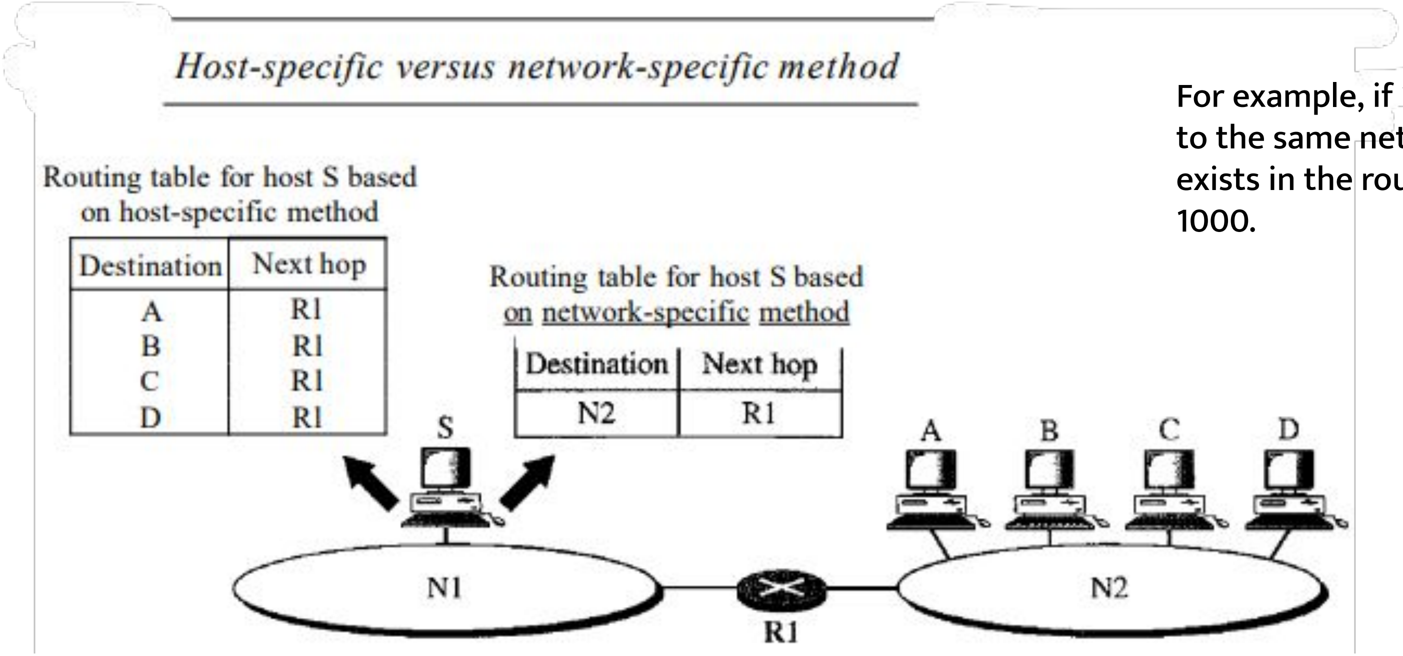
Destination	Route
Host B	Host B

Destination	Next hop
Host B	---

Computer Networks

2. Host-Specific Method Versus **Network-Specific Method** :

A second technique to reduce the routing table content and simplify the searching process is called the **network-specific method**. Here, instead of having an entry for every destination host connected to the same physical network (host-specific method), we have only one entry that defines the address of the **destination network** itself



Host-specific versus network-specific method

Routing table for host S based on host-specific method

Destination	Next hop
A	R1
B	R1
C	R1
D	R1

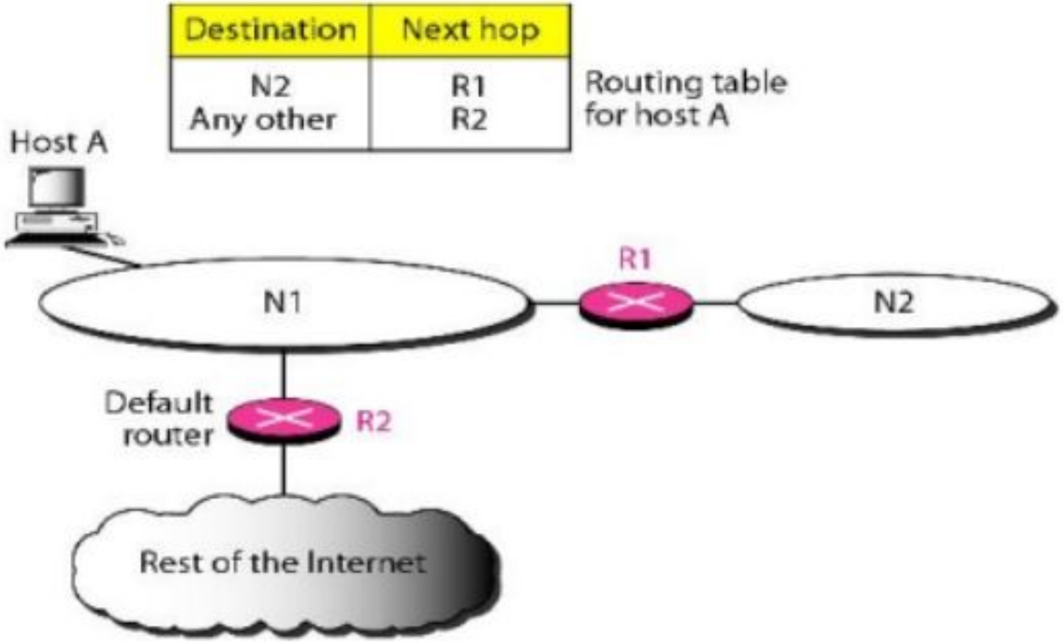
Routing table for host S based on network-specific method

Destination	Next hop
N2	R1

For example, if 1000 hosts are attached to the same network, only one entry exists in the routing table instead of 1000.

3. Default Method : Another technique to simplify routing is called the default method. Here a host sends all the packets going out of a network to a specific router called the default router.

In the figure below, host A is connected to a network with two routers. Router R1 routes the packets to hosts connected to network N2. However, for the rest of the Internet, router R2 is used. So instead of listing all networks in the entire Internet, host A can just have one entry called the default (normally defined as network address 0.0.0.0).



Destination	Next hop
N2	R1
Any other	R2

Routing table for host A

ROUTING

Network routing is the process of selecting a path across one or more networks.

Routing

Steps for routing a packet from a source to a destination.

1. Source sends the first packet to the nearest router.
2. Router receives packet
3. Examines the destination IP - When the router receives a packet, it looks at its IP header. The most important field in this header is the destination IP address, which tells the router where the packet is to be sent.
4. Router looks up in routing table for IP destination network
5. Forwards the packet through the interface to the destination network.

There are two types of Routing - **Unicast Routing** & **Multicast Routing**

Routing from one source to one destination is called **Unicast Routing** and routing from one source to multiple destination is called **Multicast Routing**

Routing Table :

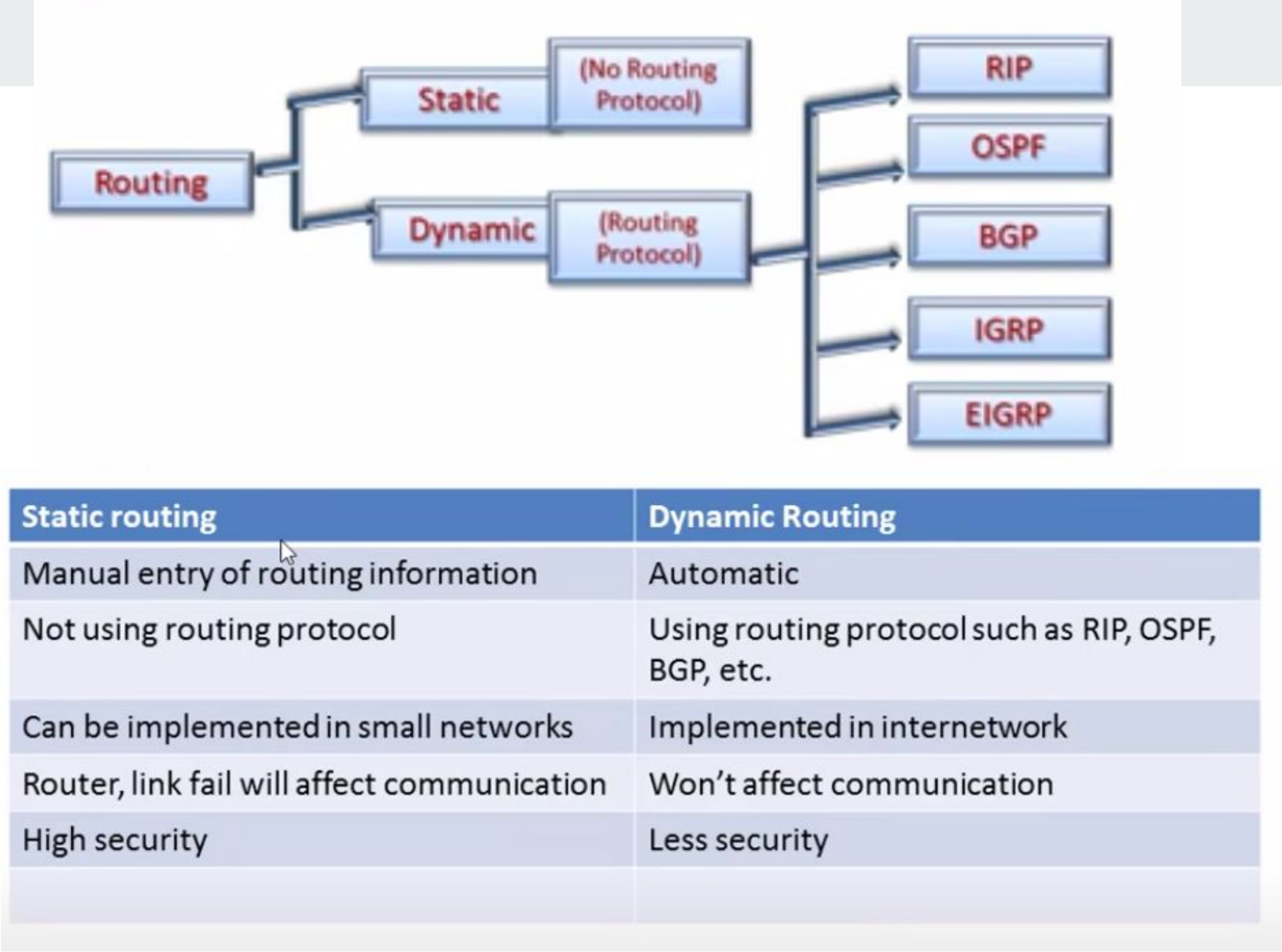
- **A routing table contains the information necessary to forward a packet along the best path towards its destination.**
- Each packet contains information about its source and destination. When a packet is received, a network device examines the packet and matches it to the routing table entry providing the best match for its destination. The table then provides the device with instructions for sending the packet to the next hop on its route across the network.
- **A router has a routing table with an entry for each destination, or a combination of destinations, to route IP packets.**
- The routing table can be either **static** or **dynamic**.
- A static table is one with manual entries. A dynamic table is the one that is updated automatically when there is a change in the network.

Static Routing Table

- A static routing table contains information entered manually.
- The administrator enters the route for each destination into the table.
- It cannot update automatically when there is a change in the network.
- If a router fails, alternate routes can’t be decided immediately. The table must be manually altered by the administrator.
- A static routing table can be used in a small network that does not change very often, or in an experimental network for troubleshooting.
- It is poor strategy to use a static routing table in a big network such as the Internet.
- No routing protocols are used.

Dynamic Routing Table

- A dynamic routing table is updated periodically by using one of the dynamic routing protocols such as RIP, OSPF, or BGP.
- Whenever there is a change in the Internet, such as a shutdown of a router or breaking of a link, the dynamic routing protocols update all the tables in the routers automatically.
- The routers in a big network such as the Internet need to be updated dynamically for efficient delivery of the IP packets.



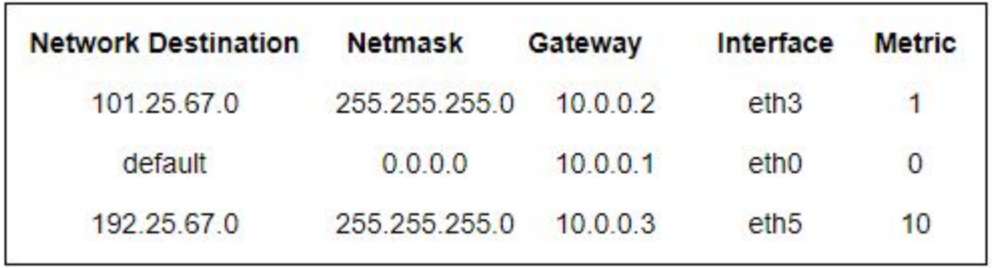
Static routing	Dynamic Routing
Manual entry of routing information	Automatic
Not using routing protocol	Using routing protocol such as RIP, OSPF, BGP, etc.
Can be implemented in small networks	Implemented in internetwork
Router, link fail will affect communication	Won't affect communication
High security	Less security

Entries in an IP Routing Table :

An IP routing table comprises of a set of rules that are used to determine where the data packets are travelling over an IP (Internet Protocol) network. All kinds of Internet Protocol-enabled devices, comprising switches and routers, utilize the routing tables.

The basic components of each entry in the routing table are:

- Network address : This field defines the network address to which the packet is finally delivered. In the case of host-specific routing, this field defines the address of the destination host.
- Subnet mask: This field defines the mask applied for the entry.
- Next Hop/Gateway: This field defines the address of the next-hop router to which the packet is delivered
- Interface: this refers to the outgoing interface that connects to the destination. Remember that every router has at least two interfaces.
- Metric: the metric is the number of hops or number of routers to be crossed to get to the destination network. If multiple routes exist, the route with the lowest metric is usually chosen.



Network Destination	Netmask	Gateway	Interface	Metric
101.25.67.0	255.255.255.0	10.0.0.2	eth3	1
default	0.0.0.0	10.0.0.1	eth0	0
192.25.67.0	255.255.255.0	10.0.0.3	eth5	10

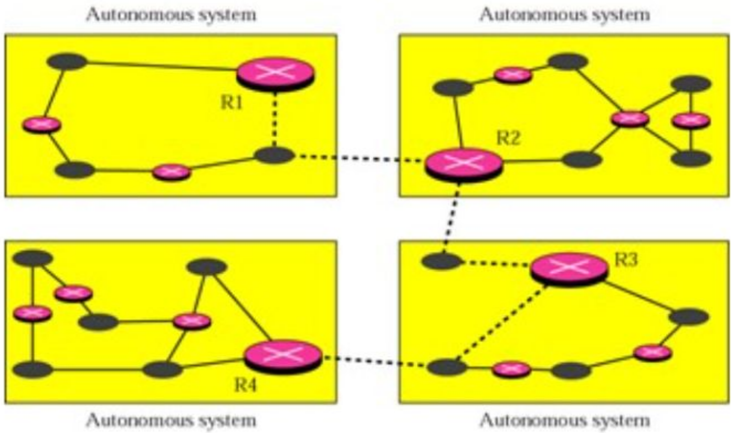
Types of Routing

There are two types of Routing

- **Intradomain Routing**
- **Interdomain Routing**

An autonomous system is a group of networks & routers under the authority of a single administrator. An internet is divided into autonomous systems. Routing inside an autonomous system is referred to as **Intradomain Routing.** Routing between autonomous systems are referred to as **Interdomain Routing**.

Each autonomous system can choose one or more intra domain routing protocols to handle routing inside the autonomous system. But only one interdomain routing protocol handles routing between autonomous systems

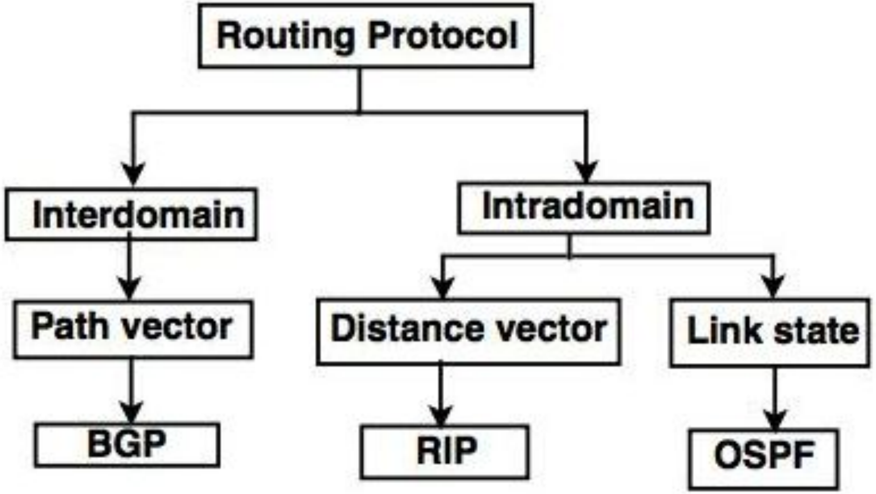


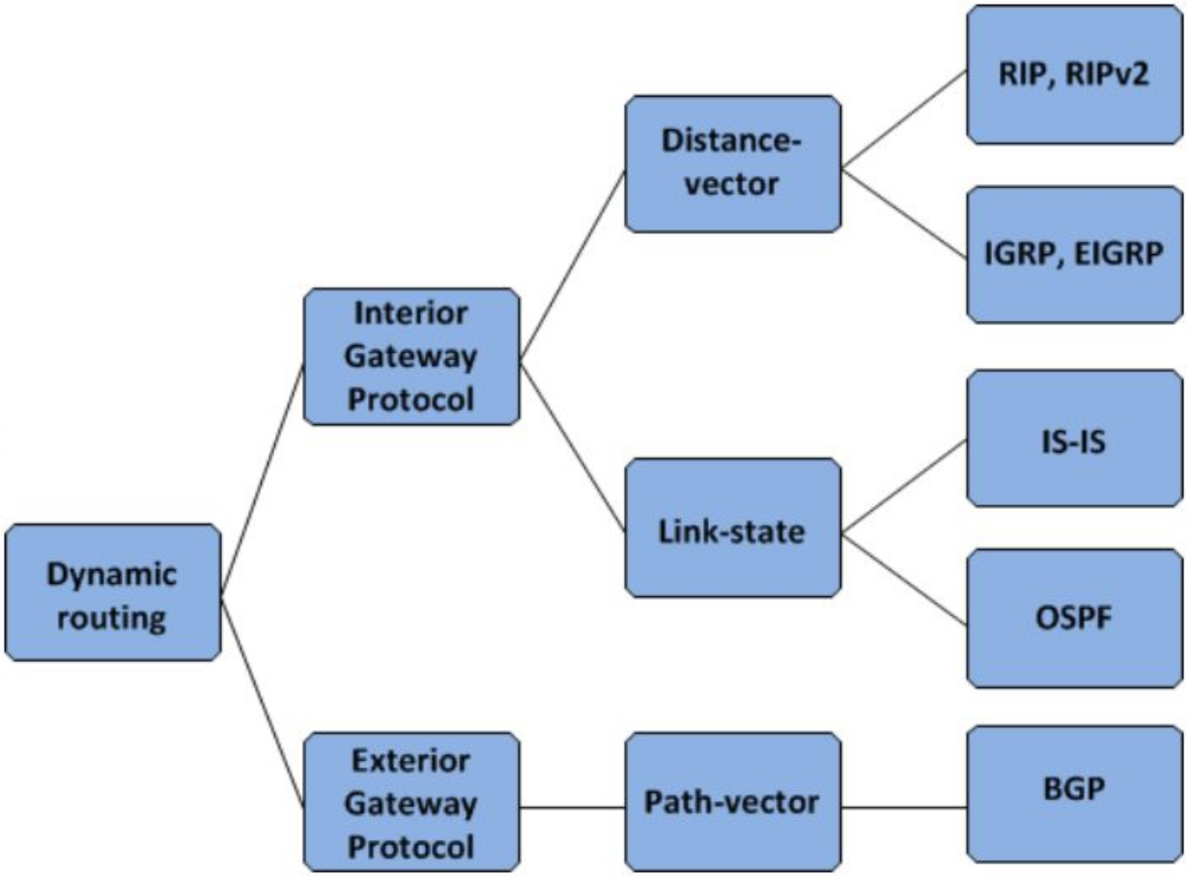
Routing Protocols : Routing protocols are special protocols used by routers to speak to each other in order to share what information they have. i.e. The routing information is exchanged between routers so that routing decisions can be made.

- Dynamic routing is implemented using Routing protocols.
- Routing Protocols can be categorized into
  - Intradomain Routing Protocols (Interior Gateway Protocols- IGP)
  - Interdomain Routing Protocols (Exterior Gateway Protocols -EGP)

- Intradomain Routing Protocols : **Distance vector** & **Link state Routing Protocols.**
- Interdomain Routing Protocol : **PathVector Routing Protocol**
- **Routing Information Protocol (RIP)** is an implementation of the Distance Vector protocol
- **Open Shortest Path First (OSPF)** is an implementation of the Link State protocol.
- **Border Gateway Protocol (BGP)** is an implementation of the Path vector protocol

Classification of routing protocol




Dynamic routing
Interior Gateway Protocol
Exterior Gateway Protocol
Distance-vector
Link-state
Path-vector
RIP, RIPv2
IGRP, EIGRP
IS-IS
OSPF
BGP

Distance Vector Routing (DVR)

- In DVR, the least cost route between any two nodes is the route with minimum distance.
- In this protocol, each node maintains a vector(table) of minimum distances to every other node.
- The table at each node also guides the packet to the desired node by showing the next hop in the routing.
- Each node shares its routing table with its immediate neighbors periodically and when there is a change.
- The information received from the neighbor router is used to update its own routing table.
- Each node can know only the distance between itself and its immediate neighbours, those are directly connected to it. They don’t allow a router to have much information about the world outside of their own neighbours.
- Distance vector routing uses UDP Protocol for transportation.

DISTANCE VECTOR ROUTING

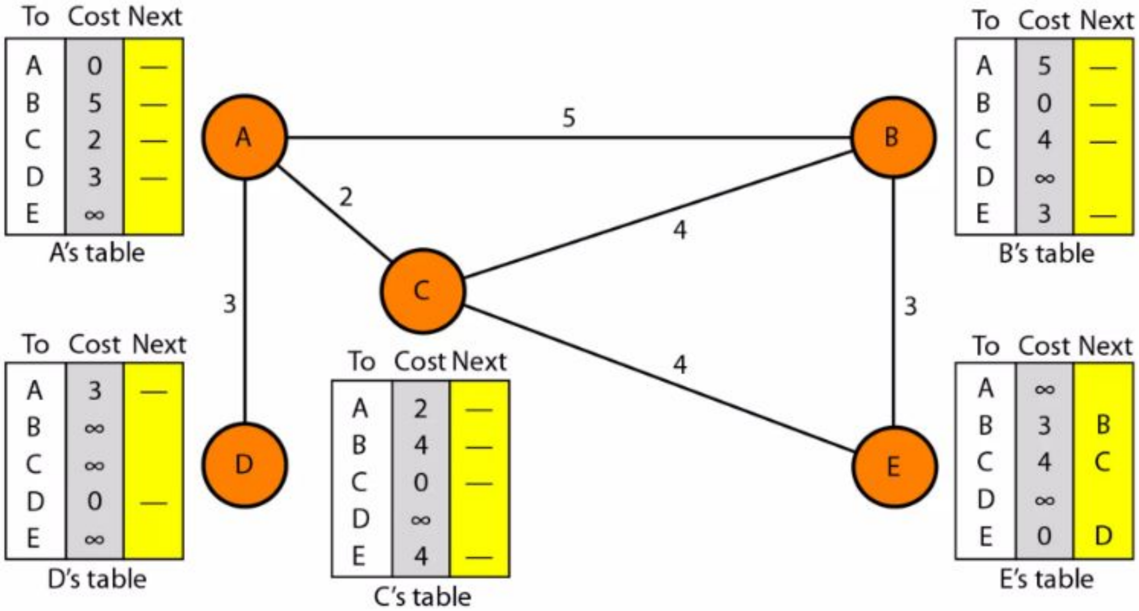
- In this protocol, as the name implies, each node maintains a vector (table) of minimum distances to every node.

Initialization

At the beginning, each node sends a message to the immediate neighbour and find the distance between them. Each node can know only the distance between itself and its immediate neighbors, those directly connected to it. The distance for any entry that is not a neighbor is marked as infinite (unreachable).

Sharing : The whole idea of distance vector routing is the sharing of information between neighbors. In distance vector routing, each node shares its routing table with its immediate neighbors periodically and when there is a change.

Updation : Whenever a node receives a table from the neighbor, it updates its own routing table.



To	Cost	Next
A	0	—
B	5	—
C	2	—
D	3	—
E	∞	

A's table

To	Cost	Next
A	5	—
B	0	—
C	4	—
D	∞	
E	3	—

B's table

To	Cost	Next
A	3	—
B	∞	
C	∞	
D	0	—
E	∞	

D's table

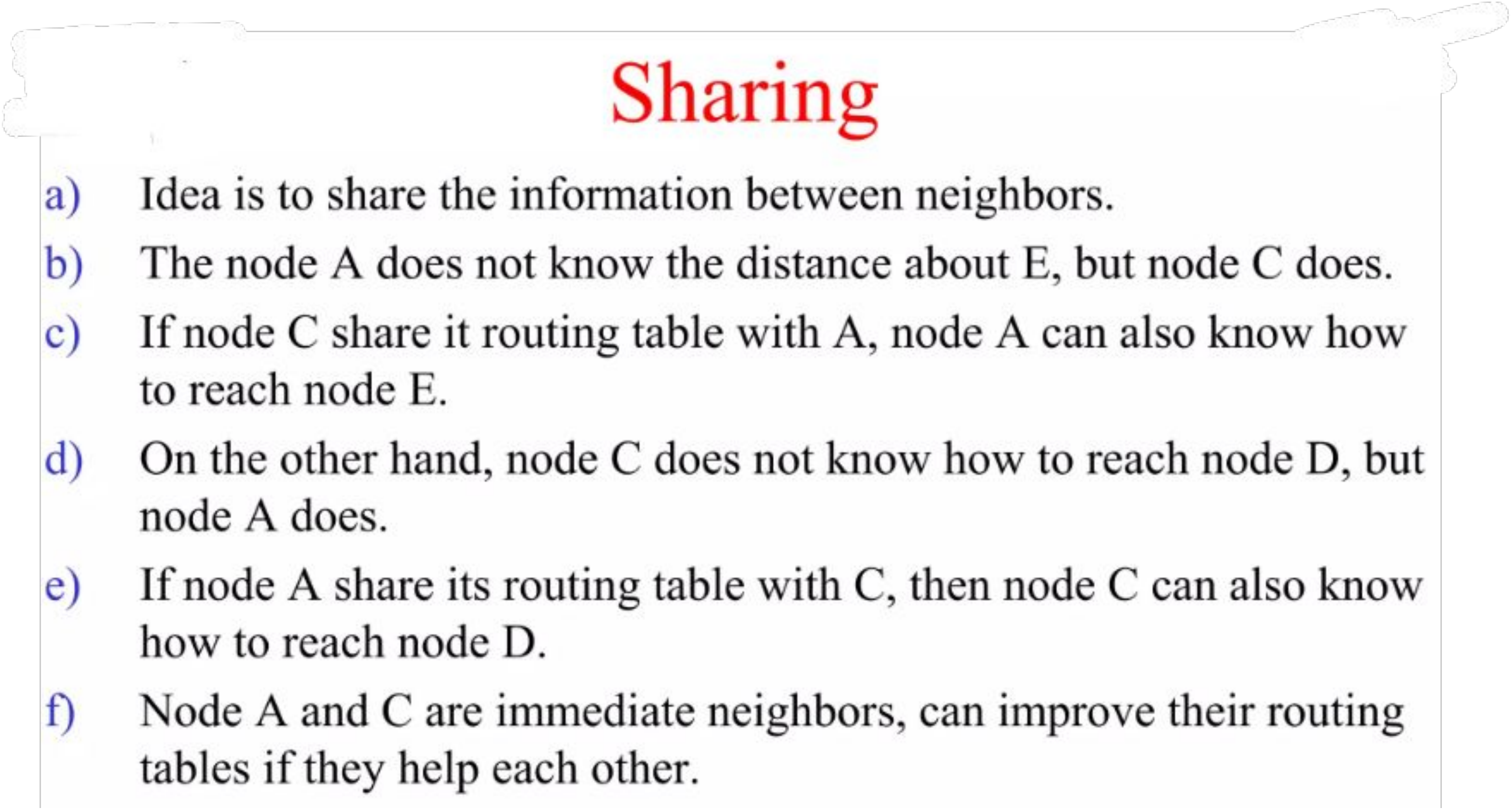
To	Cost	Next
A	2	—
B	4	—
C	0	—
D	∞	
E	4	—

C's table

To	Cost	Next
A	∞	
B	3	B
C	4	C
D	∞	
E	0	D

E's table

At the beginning, each node can only find the distance between itself and its immediate neighbours.



Sharing

a) Idea is to share the information between neighbors.

b) The node A does not know the distance about E, but node C does.

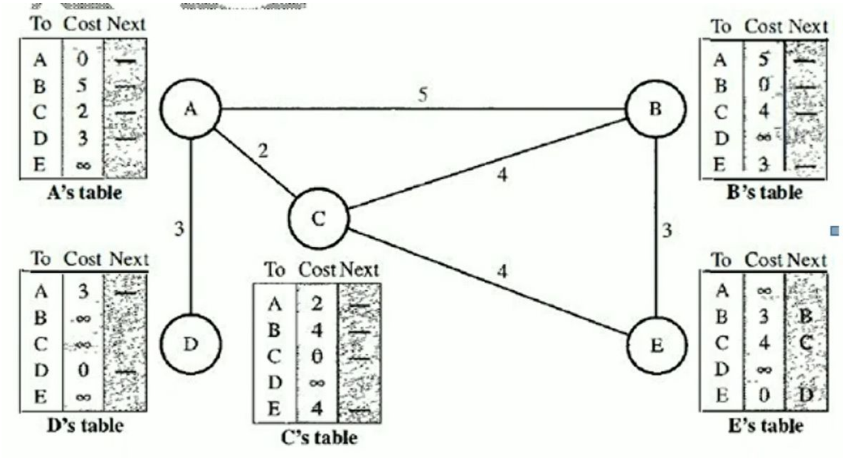
c) If node C share it routing table with A, node A can also know how to reach node E.

d) On the other hand, node C does not know how to reach node D, but node A does.

e) If node A share its routing table with C, then node C can also know how to reach node D.

f) Node A and C are immediate neighbors, can improve their routing tables if they help each other.

Updating in Distance Vector Routing



To	Cost	Next
A	0	—
B	5	—
C	2	—
D	3	—
E	∞	

A's table

To	Cost	Next
A	5	—
B	0	—
C	4	—
D	∞	
E	3	—

B's table

To	Cost	Next
A	3	—
B	∞	
C	∞	
D	0	—
E	∞	

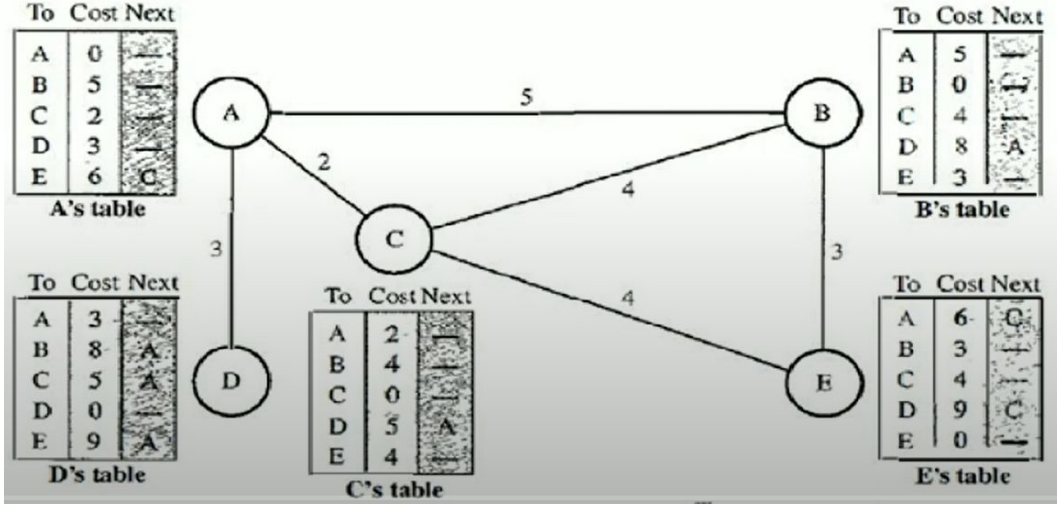
D's table

To	Cost	Next
A	2	—
B	4	—
C	0	—
D	∞	
E	4	—

C's table

To	Cost	Next
A	∞	
B	3	B
C	4	C
D	∞	
E	0	D

E's table



To	Cost	Next
A	0	—
B	5	—
C	2	—
D	3	—
E	6	C

A's table

To	Cost	Next
A	5	—
B	0	—
C	4	—
D	8	A
E	3	—

B's table

To	Cost	Next
A	3	—
B	8	A
C	5	A
D	0	—
E	9	A

D's table

To	Cost	Next
A	2	—
B	4	—
C	0	—
D	5	A
E	4	—

C's table

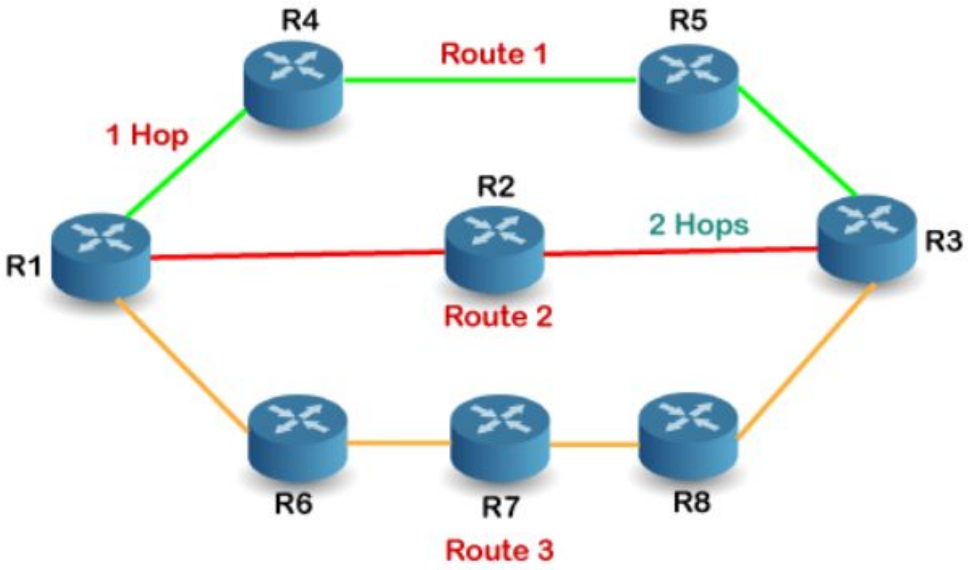
To	Cost	Next
A	6	C
B	3	—
C	4	—
D	9	C
E	0	—

E's table

RIP - Routing Information Protocol

- RIP stands for Routing Information Protocol. RIP is an intra-domain routing protocol used within an autonomous system.
- RIP is based on the **distance vector-based** strategy, so we consider the entire structure as a graph where nodes are the routers, and the links are the networks.
- In a routing table, the first column is the destination. The cost metric is the number of hops to reach the destination. The number of hops available in a network would be the cost. Hop count is the number of routers occurring in between the source and destination network. The path with the lowest hop count is considered as the best route to reach a network
- **Therefore, It is a dynamic routing protocol that uses hop count as a routing metric to find the best path between the source and the destination network.**
- In RIP, infinity is defined as 16, which means that the RIP is useful for smaller networks or small autonomous systems. The maximum number of hops that RIP can contain is 15 hops, i.e., it should not have more than 15 hops as 16 is infinity.
- Routers running the distance-vector protocol send all or a portion of their routing tables in routing-update messages to their neighbors.
- RIP is used to configure the hosts as part of a network. This type of routing requires little maintenance and also automatically re-configures routing tables when your network changes

How does the RIP work?



If there are 8 routers in a network where Router 1 wants to send the data to Router 3. If the network is configured with RIP, it will choose the route which has the least number of hops. There are three routes in the above network, i.e., Route 1, Route 2, and Route 3. The Route 2 contains the least number of hops, i.e., 2 where Route 1 contains 3 hops, and Route 3 contains 4 hops, so RIP will choose Route 2.

Link State Routing (LSR) Algorithm

The Link State Routing (LSR) algorithm, also known as the **Dijkstra algorithm**, is a routing protocol used in computer networks to determine the shortest path between nodes.

Link state routing is an interior protocol that updates the routers inside the autonomous system i.e it is an intradomain protocol. The link-state routing was introduced to overcome the shortcomings of the distance vector routing protocol.

In this algorithm, each node in the network maintains a local database of the topology of the network, which is updated whenever changes occur. The node then calculates the shortest path to all other nodes in the network using this information. The result is then flooded to all other nodes in the form of Link State Packets (LSPs). This information is then used by each node to update its own routing table, which it uses to determine the next hop for each destination. The LSR algorithm is commonly used in large enterprise networks and in Open Shortest Path First (OSPF) routing protocol.

In link state routing, four sets of actions are required to ensure that each node has the routing table showing the least-cost node to every other node.

The working of link-state routing can be stated in four simple steps.

1. Creation of the states of the links by each node, called the link state packet(LSP).
2. Advertise LSP containing its link-state information to all other routers in the domain, which is called **flooding**. Also, receives information from other routers.
3. Formation of shortest path tree for each node.
4. Calculation of a routing table based on the shortest path tree

Comparing with DVR, in DVR, we circulate the entire routing table to the nearest neighbours, where as in LSR, we will sent only the link state information to the nearest router.

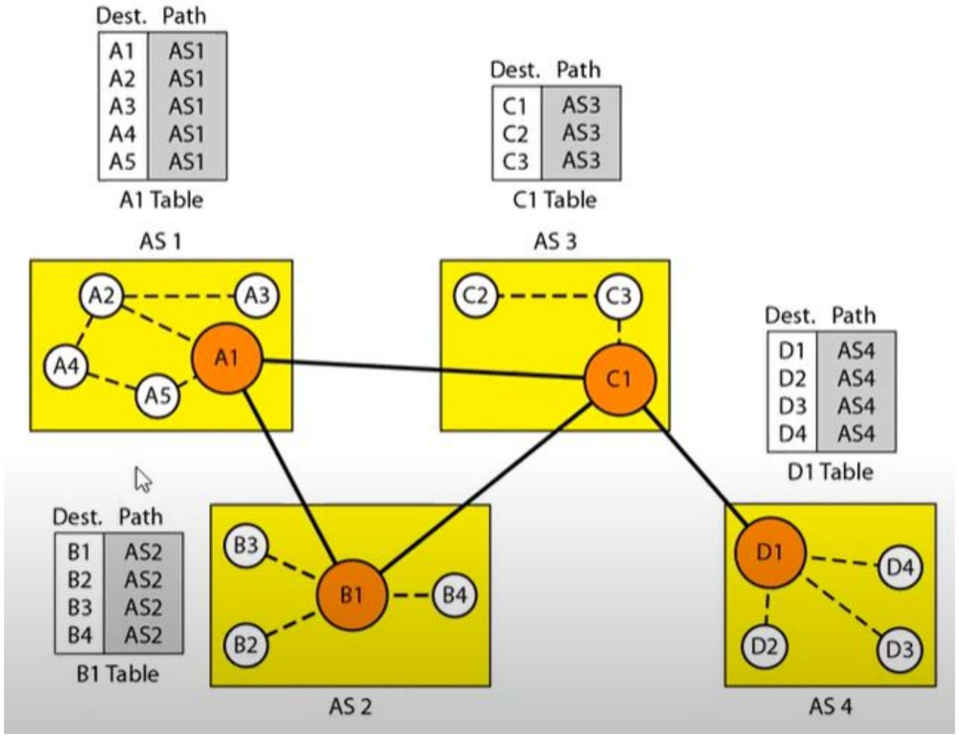
OSPF (Open Shortest Path First)

- The OSPF protocol is a **Link-State Routing protocol,** which means that the routers exchange topology information with their nearest neighbors.
- The topology information is flooded throughout the Autonomous System(AS), so that every router within the AS has a complete picture of the topology of the AS.
- This picture is then used to calculate end-to-end paths through the AS, using the Dijkstra algorithm.
- Therefore, in a link-state routing protocol, the next hop address to which data is forwarded is determined by choosing the best end-to-end path to the destination.
- The main advantage of a link state routing protocol like OSPF is that the complete knowledge of topology allows routers to calculate routes that satisfy particular quality of service requirements.

Flooding : Flooding is the forwarding of a packet by a router from one node to every other node attached to the router except the one from which the packet arrived. It is a way to distribute routing information updates quickly to every node in a large network.

Path Vector Routing

- Path vector routing is an interdomain routing protocol.
- In path vector routing there is a node in each autonomous system that acts on behalf of the entire autonomous system. That node is known as the **speaker node**.
- The speaker node in an AS creates a routing table and advertises it to speaker nodes in the neighbouring Autonomous Systems.
- The is the same as for distance vector routing except that only speaker nodes in each AS can communicate with each other.
- A speaker node advertises the path, not the metric of the nodes in its autonomous system or other autonomous system.
- Therefore, a path-vector routing protocol is **a network routing protocol which maintains the path information that gets updated dynamically**.
- **BORDER GATEWAY PROTOCOL (BGP)** is an interdomain routing protocol using path vector routing.



Dest.	Path
A1	AS1
A2	AS1
A3	AS1
A4	AS1
A5	AS1

A1 Table

Dest.	Path
C1	AS3
C2	AS3
C3	AS3

C1 Table

Dest.	Path
D1	AS4
D2	AS4
D3	AS4
D4	AS4

D1 Table

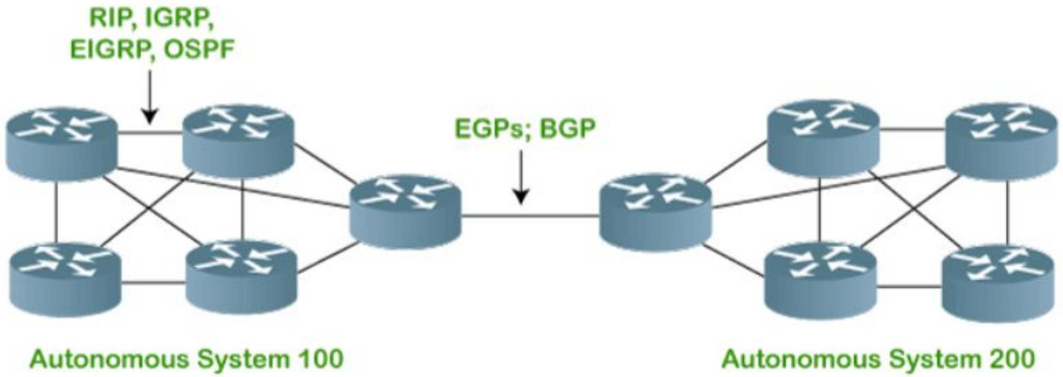
Dest.	Path
B1	AS2
B2	AS2
B3	AS2
B4	AS2

B1 Table

Border Gateway Protocol

Border Gateway Protocol (BGP) is an interdomain routing protocol, and it uses the path-vector routing. It is standardized Exterior Gateway Protocol that is used to exchange routing information among the autonomous system on the internet.

It is designed to exchange routing information between routers in different autonomous systems (AS) on the Internet. It is used to connect individual networks into the larger Internet and establish routing policies. BGP is an essential component of the Internet routing infrastructure and is used to build and maintain large-scale networks.

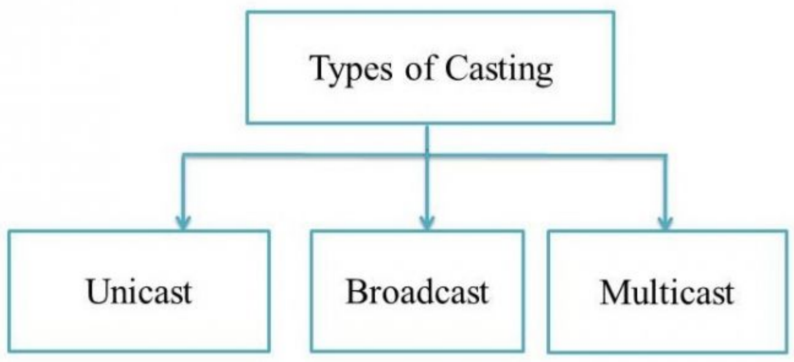


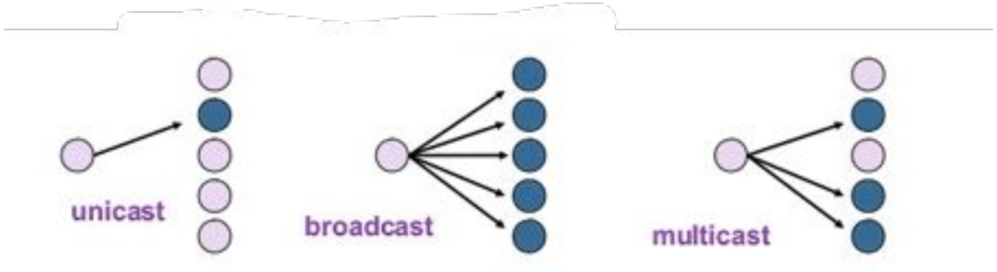
Difference between Unicast, Broadcast and Multicast in Computer Network

Casting in computer networks means transmitting data (stream of packets) over a network.

1. Unicast (One-to-One)

In Unicast transmission, the data is transferred from a single sender to a single receiver). So, in short, you can term it as a one-to-one transmission.
For example, if a device having IP address 10.1.2.0 in a network wants to send the traffic stream(data packets) to the device with IP address 20.12.4.2 in the other network, then unicast comes into the picture. This is the most common form of data transfer over the networks.





Broadcast Transmission (One-to-All)

In Broadcast transmission, the data is transmitted from one sender to all the receivers within the same network or in other networks. This type of transmission is used in ARP (Address Resolution Protocol), RARP and RIP (Routing Information Protocol) where all the devices must see the data.

There are two types of broadcast transmission –

- Limited Broadcast
- Directed Broadcast

Limited Broadcast : In Limited Broadcast, the data is transmitted from a single source host to all the other hosts residing in the same network.
Directed Broadcast : Directed Broadcast transmits data from one source host to all the other hosts that exist in some other network.

Multicast Transmission (One-to-Many)

When the data is transmitted from a single source host to a specific group of hosts having the interest to receive the data, it is known as multicast transmission. Multicast can be more efficient than unicast when different groups of receivers need to see the same data.

Multicast Routing

Multicast routing is a method of delivering data from a single source to multiple recipients simultaneously over a computer network. It enables efficient and scalable distribution of information to multiple receivers, reducing network congestion and reducing the amount of data transmitted across the network.

In multicast routing, a single source sends data to a designated multicast group address, which represents a set of receivers interested in receiving the same data. A multicast source, such as a live video conference, sends traffic in one stream to a multicast group. The multicast group contains receivers such as computers, devices, and IP phones. Routers in the network use multicast routing protocols, such as PIM (Protocol Independent Multicast), to forward the data only to the network segments where there are receivers, thereby reducing network traffic and increasing network efficiency.

Multicast routing is commonly used in technologies such as:

- Voice over IP (VOIP)
- Video conferencing
- IP television (IPTV)
- Online gaming
- Financial data distribution

Transport layer

At the sender's side: The transport layer receives data (message) from the Application layer and then performs Segmentation, divides the actual message into segments, adds source and destination port numbers into the header of the segment, and transfers the message to the Network layer.

At the receiver's side: The transport layer receives data from the Network layer, reassembles the segmented data, reads its header, identifies the port number, and forwards the message to the appropriate port in the Application layer.

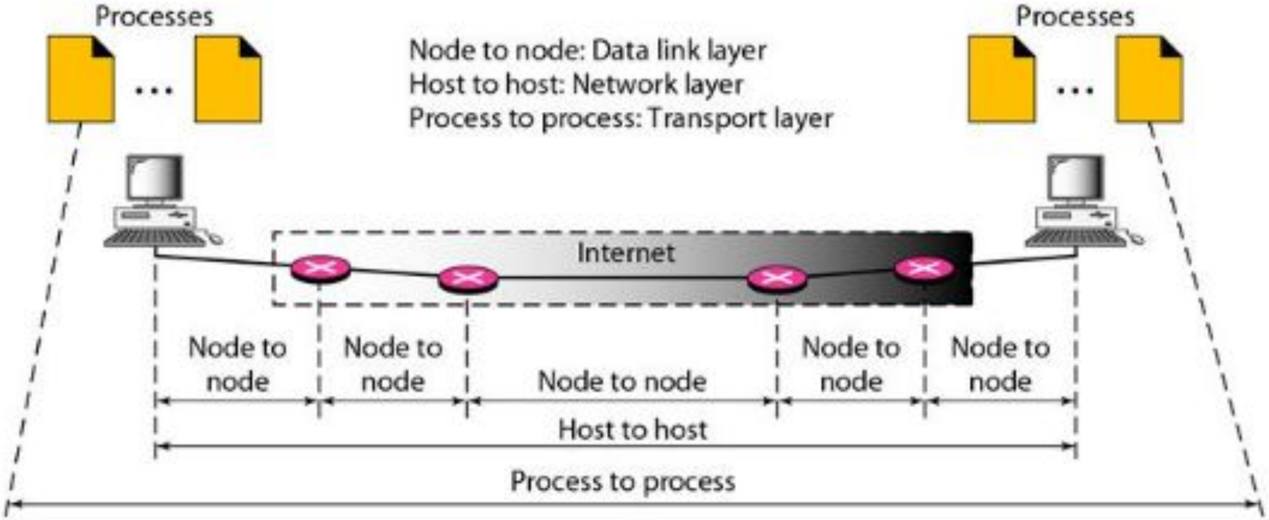
The functions of the transport layer

Segmentation & reassembly : A message is divided into transmittable segments , with each segment containing a sequence number. These numbers enable the transport layer to reassemble the message correctly upon arriving at the destination and also to identify and replace packets that were lost in transmission.

Service-point Addressing : Computers often run several programs at the same time. The transport layer header must therefore include a type of address called a service-point address (port address). The network layer gets each packet to the correct computer, and the transport layer gets the entire message to the correct process on that computer. i.e. Transport layer is responsible for process to process delivery of the entire message.

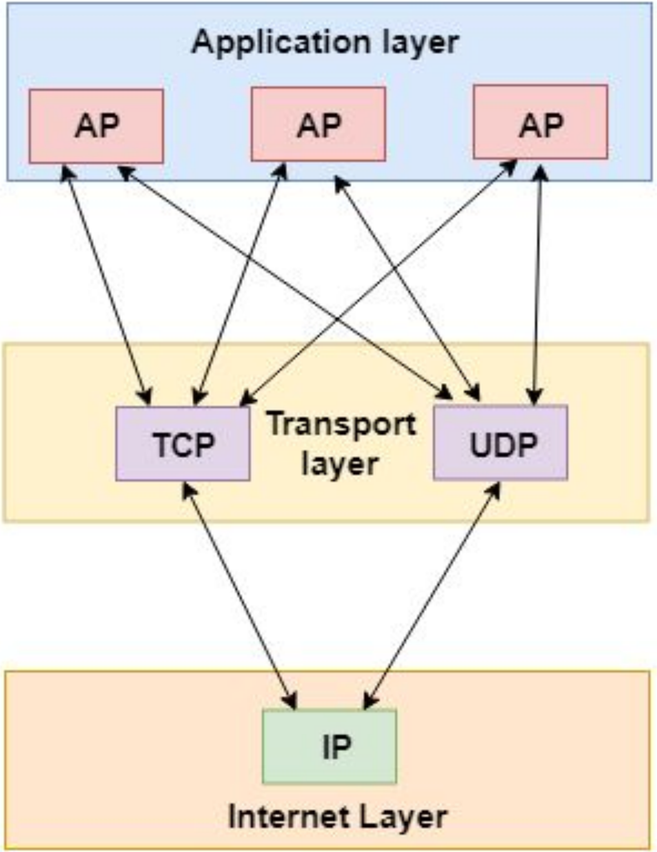
PROCESS-TO-PROCESS DELIVERY :

The data link layer is responsible for delivery of frames between two neighboring nodes over a link. This is called node-to-node delivery. The network layer is responsible for delivery of datagrams between two hosts. This is called host-to-host delivery. Real communication takes place between two processes (application programs). The transport layer is responsible for process-to-process delivery- i.e. the delivery of a packet, part of a message, from one process to another.



Connection control : The transport layer can be either connectionless or connection-oriented. A connectionless transport layer treats each segment as an independent packet and delivers it to the transport layer at the destination machine. A connection-oriented transport layer makes a connection with the transport layer at the destination machine first before delivering the packets. After all the data is transferred, the connection is terminated. UDP is a connectionless protocol whereas TCP is a connection oriented protocol.

Flow Control : Like data link layer, the transport layer is responsible for flow control. However, flow control in this layer is performed end-to-end rather than across node to node. TCP prevents data loss due to a fast sender and slow receiver by imposing some flow control techniques. It uses the method of sliding window protocol which is accomplished by the receiver by sending a window back to the sender informing the size of data it can receive. UDP, is not involved in flow control.



Error Control : Like data link layer, the transport layer is responsible for error control. However, error control in this layer is performed end-to-end rather than across node to node. The sending transport layer makes sure that the entire message arrives at the receiving transport layer without error(damage, loss or duplication). Error correction is usually achieved through retransmission. Error detection and correction techniques used in transport layer is checksum method.

Multiplexing and Demultiplexing: Multiplexing allows simultaneous use of different applications over a network that is running on a host. The transport layer provides this mechanism which enables us to send packet streams from various applications simultaneously over a network. The transport layer accepts these packets from different processes differentiated by their port numbers and passes them to the network layer after adding proper headers. Similarly, Demultiplexing is required at the receiver side to obtain the data coming from various processes. Transport receives the segments of data from the network layer and delivers it to the appropriate process running on the receiver’s machine.

User Datagram Protocol (UDP)

- The User Datagram Protocol (UDP) is called a connectionless, unreliable transport protocol.
- The UDP is a lightweight data transport protocol that works on top of IP.
- It does not add anything to the services of Internet Protocol except to provide process-to-process communication instead of host-to-host communication.
- UDP performs very limited error checking in packets, and does not attempt to solve other problems that arise with packets, such as lost or out of order packets. That's why UDP is sometimes known as the Unreliable Data Protocol. .

UDP is simple but fast, at least in comparison to other protocols that work over IP. It is often used for time-sensitive applications (such as real-time video streaming) where speed is more important than accuracy. In UDP, the receiver does not generate an acknowledgement of packet received and in turn, the sender does not wait for any acknowledgement of packet sent. This shortcoming makes this protocol unreliable as well as easier on processing.

Why do we need UDP, an unreliable protocol to transport the data?
In case of video streaming, thousands of packets are forwarded towards its users. Acknowledging all the packets is troublesome and may contain huge amount of bandwidth wastage. The best delivery mechanism of underlying IP protocol ensures best efforts to deliver its packets, but even if some packets in video streaming get lost, the impact is not disastrous and can be ignored easily. Loss of few packets in video and voice traffic sometimes goes unnoticed.

User Datagram Format:

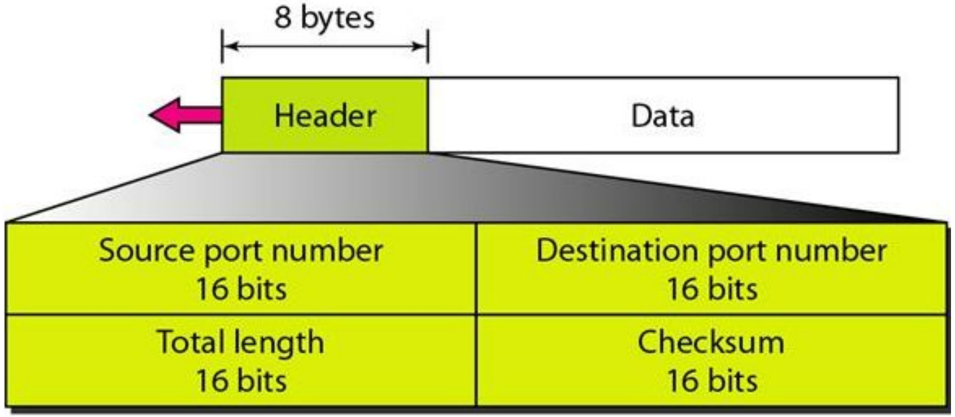
UDP packets are called user datagrams. They have a fixed-size header of 8 bytes. The figure shows the format of a user datagram.

The fields are as follows:

- **Source port number:** This is the port number used by the process running on the source host. It is 16 bits long, which means that the port number can range from 0 to 65,535.

- **Destination port number:** This is the port number used by the process running on the destination host. It is also 16 bits long.

- **Length:** This is a 16-bit field that defines the total length of the user datagram, i.e. header plus data.



8 bytes

Header

Data

Source port number
16 bits

Destination port number
16 bits

Total length
16 bits

Checksum
16 bits

- **Checksum:** This 16-bit field is used to detect errors over the entire user datagram (header plus data).

Uses of UDP

The following lists some uses of the UDP protocol:

- UDP results in faster communication because it does not spend time forming a firm connection with the destination before transferring the data. Because establishing the connection takes time, eliminating this step results in faster data transfer speeds.
- UDP is suitable for a process that requires simple communication (one request-one response) with little concern for flow and error control. Eg: DNS (Domain Name Server) which returns the IP address of a URL.
- Since the header size of UDP is very small, its over head is very less and hence applications using UDP will work faster. Eg: online games, Voice over IP
- It is specifically chosen for time-sensitive applications like gaming, playing videos, or Domain Name System (DNS) lookups.
- UDP is a suitable transport protocol for multicasting/ broadcasting.
- It is used for some route updating protocols such as Routing Information Protocol (RIP). Here every node will share its routing information to all other nodes in fixed intervals. If TCP is used, TCP will have to establish connection which requires to reserve large buffer area.
- Applications like Skype, Youtube etc which requires continuous streaming uses UDP.

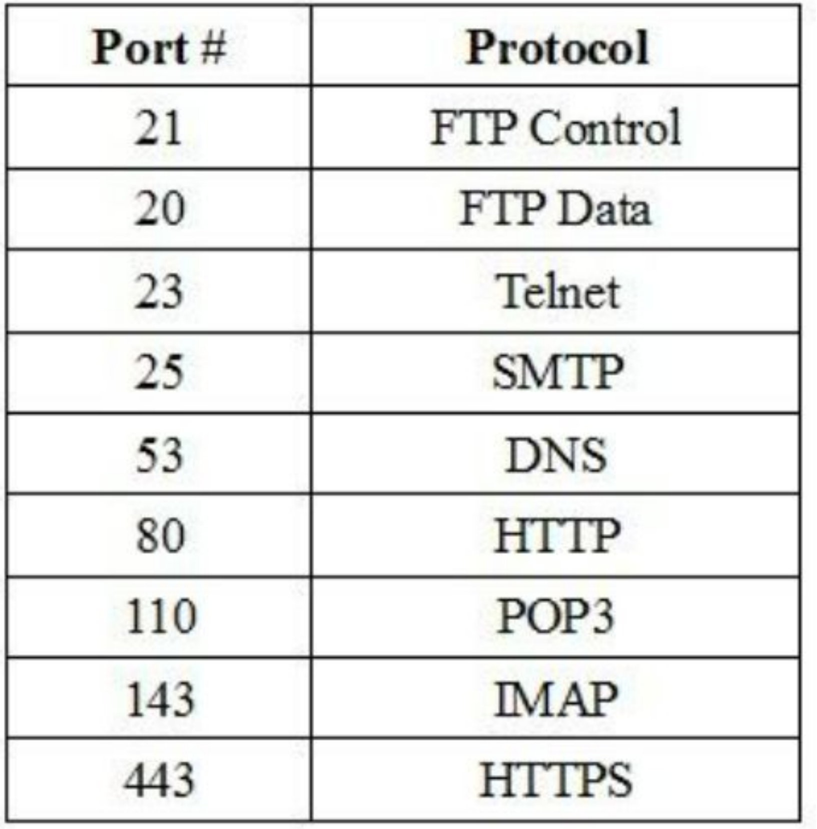
TCP - Transmission Control Protocol

- The Transmission Control Protocol (TCP) is one of the most important protocols of Internet Protocols suite. It is most widely used protocol for data transmission in communication network such as internet.
- It is a transport protocol that is used on top of IP to ensure reliable transmission of packets.
- TCP, like UDP, is a process-to-process protocol. Therefore, it uses port numbers in its header.
- Unlike UDP, TCP is a connection oriented protocol; i.e. it creates a virtual connection between two hosts to send data.
- TCP includes mechanisms to solve many of the problems that arise from packet-based messaging, such as lost packets, out of order packets, duplicate packets, and corrupted packets.
- TCP is reliable protocol. That is, the receiver always sends either positive or negative acknowledgement about the data packet to the sender, so that the sender gets information whether the data packet is reached the destination or it needs to be resend.
- TCP provides error-checking and recovery mechanism. It also provides flow control and quality of service.
- In brief, TCP is called a connection-oriented, reliable transport protocol. It adds connection-oriented and reliability features to the services of IP.

Features of Transmission Control Protocol : TCP has several features that are briefly summarized below

- **Transport Layer Protocol :** TCP is a transport layer protocol as it is used in transmitting the data from the sender to the receiver.
- **Process-to-Process Communication :** TCP provides process-to-process communication using port numbers.
- **Connection-oriented :** It is a connection-oriented service that means the data exchange occurs only after the connection establishment. When the data transfer is completed, then the connection will get terminated.
- **Reliable :** TCP is a reliable transport protocol as it follows the flow and error control mechanism. It also supports the acknowledgment mechanism, which checks the arrival of the data at the receiver. In the acknowledgment mechanism, the receiver sends either positive or negative acknowledgment to the sender so that the sender can get to know whether the data packet has been received or needs to resend.
- **Segment Numbering System :** TCP keeps track of the segments being transmitted or being received by assigning numbers to each and every byte. Acknowledgment Numbers are assigned to received segments.
- **Order of the data is maintained :** This protocol ensures that the data reaches the intended receiver in the same order in which it is sent. It orders and numbers each segment so that the TCP layer on the destination side can reassemble them based on their ordering.
- **Full duplex :**TCP offers full-duplex service, in which data can flow in both directions at the same time.
- **Congestion Control :** TCP takes into account the level of congestion in the network. Congestion level is determined by the amount of data sent by a sender.

Well-known ports used by TCP



Port #	Protocol
21	FTP Control
20	FTP Data
23	Telnet
25	SMTP
53	DNS
80	HTTP
110	POP3
143	IMAP
443	HTTPS

0 to 1023 = **well known port numbers** used by the standard applications

1024 to 49151 = **Registered ports** used by vendors for their own server applications.

49152 to 65535 = **Dynamic ports**
These port numbers are used by clients and not servers. They an be used dynamically by applications and are not registered or controlled.

TCP Segment Format

A packet in TCP is called a segment. The format of a segment is shown in the following figure.

The segment consists of a 20- to 60-byte header, followed by data from the application program. The header is 20 bytes if there are no options and up to 60 bytes if it contains options. The different sections of the Header are as follows.

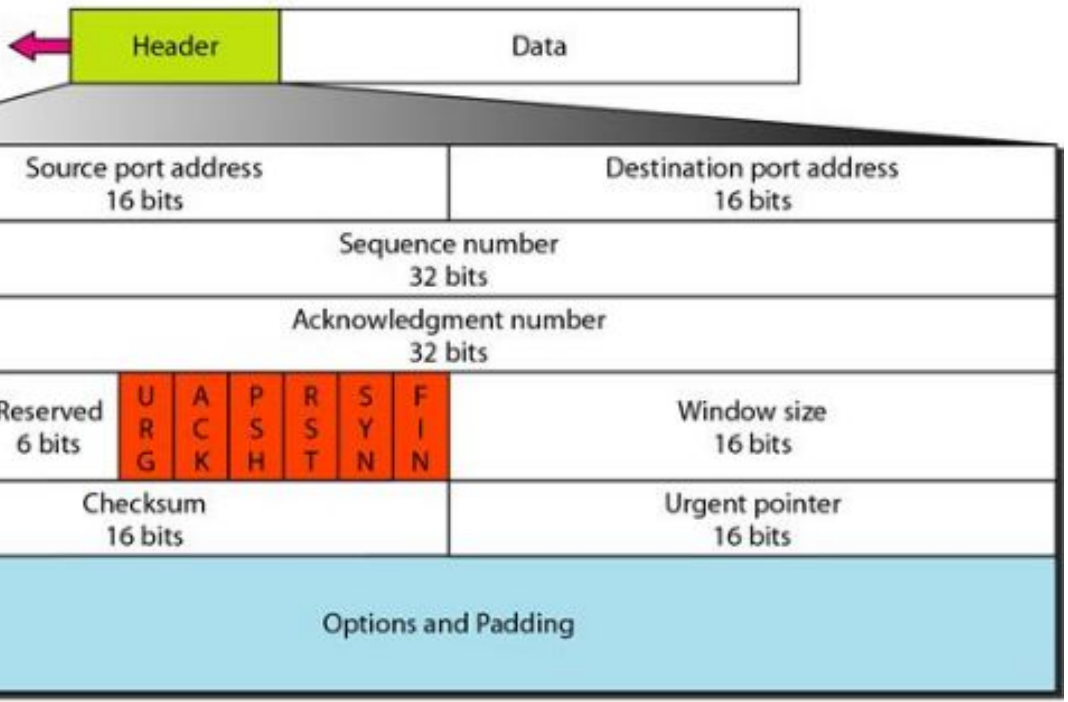
Source port address:

This is a 16-bit field that defines the port number of the application program in the host that is sending the segment.

Destination port address:

This is a 16-bit field that defines the port number of the application program in the host that is receiving the segment.

A TCP connection is uniquely identified by using-Combination of port numbers and IP Addresses of sender and receiver



Header

Data

Source port address 16 bits

Destination port address 16 bits

Sequence number 32 bits

Acknowledgment number 32 bits

HLEN 4 bits

Reserved 6 bits

URG

ACK

PSH

RST

SYN

FIN

Window size 16 bits

Checksum 16 bits

Urgent pointer 16 bits

Options and Padding

Sequence number:

- Sequence number is a 32 bit field.
- TCP assigns a unique sequence number to each byte of data contained in the TCP segment.
- This field contains the sequence number of the first data byte.

Acknowledgment number:

- Acknowledgment number is a 32 bit field.
- It contains sequence number of the data byte that receiver expects to receive next from the sender.
- It is always sequence number of the last received data byte incremented by 1.

Header length:

- Header length is a 4 bit field.
- It contains the length of TCP header.
- The length of the header can be between 20 and 60 bytes. Therefore, the value of this field can be between 5 (5 x 4 =20) and 15 (15 x 4 =60).
- It helps in knowing from where the actual data begins.

Reserved Bits: The 6 bits are reserved for future use. These bits are not used and are always set as zero.

Control:

This field defines 6 different control bits or flags as shown below and One or more of these bits can be set at a time. These bits enable flow control, connection establishment and termination, and the mode of data transfer in TCP. A brief description of each bit is as follows:

URG - It indicates the receiver that certain amount of data within the current segment is urgent.

ACK - ACK bit indicates whether acknowledgement number field is valid or not. When ACK bit is set to 1, it indicates that acknowledgement number contained in the TCP header is valid. For all TCP segments except request segment, ACK bit is set to 1. Request segment is sent for connection establishment during **Three Way Handshake**.

PSH - Push the data. When PSH bit is set to 1,

- All the segments in the buffer are immediately pushed to the receiver.
- No wait is done for filling the entire buffer.

RST - Reset the TCP connection. It indicates the receiver to terminate the connection immediately. It causes both the sides to release the connection and all its resources abnormally. This is used only when there are unrecoverable errors.

SYN - SYN bit is used to synchronize the sequence numbers. When SYN bit is set to 1, it indicates the receiver that the sequence number contained in the TCP header is the initial sequence number. Request segment sent for connection establishment during Three way handshake contains SYN bit set to 1.

FIN - FIN bit is used to terminate the TCP connection. The segment sent for **TCP Connection Termination** contains FIN bit set to 1.

Window size:

- Window size is a 16 bit field.
- It contains the size of the receiving window of the sender.
- It advertises how much data (in bytes) the sender can receive without acknowledgement.
- Thus, window size is used for Flow Control.

Checksum:

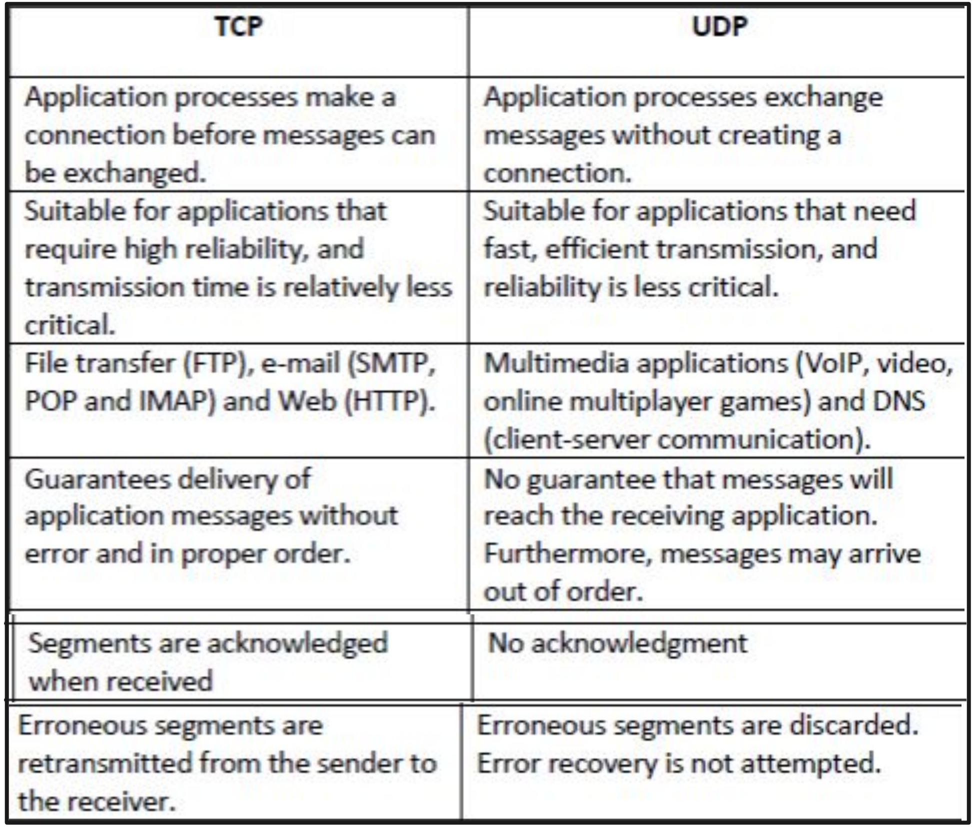
- Checksum is a 16 bit field used for error control.
- It verifies the integrity of data in the TCP payload.
- Receiver rejects the data that fails the CRC check.
- However, the inclusion of the checksum in the UDP datagram is optional, whereas the inclusion of the checksum for TCP is mandatory.

Urgent Pointer :

- Urgent pointer is a 16 bit field.
- It defines a value that will be added to the sequence number to get the sequence number of the last urgent byte.
- This field is considered valid and evaluated only if the URG bit is set to 1.

Options :

- Options field is used to provide additional options
- The size of options field vary from 0 bytes to 40 bytes.



TCP	UDP
Application processes make a connection before messages can be exchanged.	Application processes exchange messages without creating a connection.
Suitable for applications that require high reliability, and transmission time is relatively less critical.	Suitable for applications that need fast, efficient transmission, and reliability is less critical.
File transfer (FTP), e-mail (SMTP, POP and IMAP) and Web (HTTP).	Multimedia applications (VoIP, video, online multiplayer games) and DNS (client-server communication).
Guarantees delivery of application messages without error and in proper order.	No guarantee that messages will reach the receiving application. Furthermore, messages may arrive out of order.
Segments are acknowledged when received	No acknowledgment
Erroneous segments are retransmitted from the sender to the receiver.	Erroneous segments are discarded. Error recovery is not attempted.

TCP Connection

- TCP provides reliable communication. The Protocol Data Unit(PDU) of the transport layer is called a **segment**.
- A connection-oriented transport layer protocol establishes a virtual path between the source/client and the destination/server.
- All the segments belonging to the message are then sent over this virtual path.
- In TCP connection-oriented transmission requires 3 phases
  - Connection Establishment
  - Data Transfer
  - Connection Termination

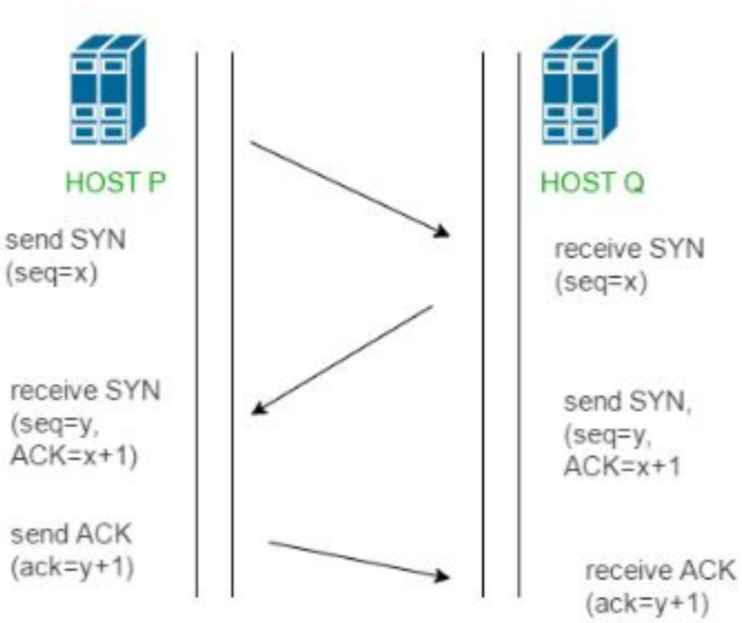
1. Connection Establishment - TCP is a connection-oriented protocol & transmits data in full-duplex mode. This implies that each party must initialize communication and get approval from the other party before any data is transferred.

In TCP, the connection is established by using **three-way handshaking**.

1. The client sends the segment with its sequence number.
2. The server, in return, sends its response with its own sequence number as well as the acknowledgement sequence, which is one more than the client sequence number.
3. When the client receives the acknowledgment of its segment, then it sends the acknowledgment to the server. In this way, the connection is established between the client and the server

TCP Connection Establishment Steps

- **Step 1 (SYN):** In the first step, the client wants to establish a connection with a server, so it sends a segment with SYN(Synchronize Sequence Number- a random no.=x) which informs the server that the client is likely to start communication
- **Step 2 (SYN + ACK):** Server responds to the client request with SYN+ACK signal bits set. Acknowledgement(ACK=x+1) signifies the response of the segment it received and SYN signifies with what sequence number=y it is likely to start the segments with
- **Step 3 (ACK):** In the final part client acknowledges the response of the server and they both establish a reliable connection with which they will start the actual data transfer



2. Data Transfer :

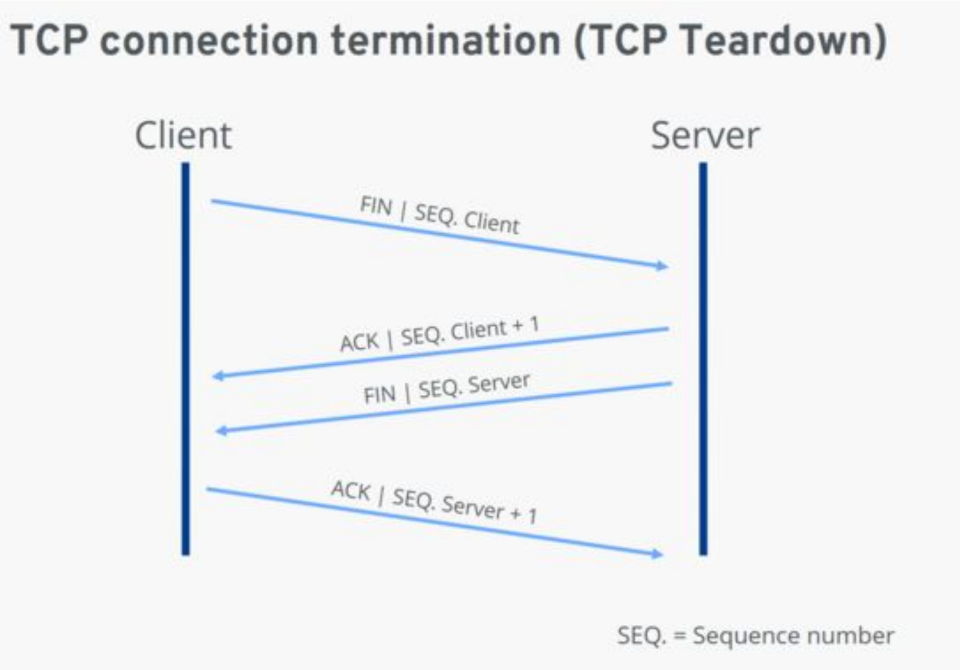
- Once the computers are done with the handshake, they're ready to receive packets containing actual data.
- When a packet of data is sent over TCP, the recipient must always acknowledge what they received.
- The first computer sends a packet with data and a sequence number. The receiver acknowledges it by setting the ACK bit and increasing the acknowledgement number.
- The sequence and acknowledgement numbers are part of the TCP header. Those two numbers help the computers to keep track of which data was successfully received, which data was lost, and which data was accidentally sent twice.
- Either computers can close the connection when they no longer want to send or receive data.

A computer initiates closing the connection by sending a packet with the FIN bit set to 1 (FIN = finish). The other computer replies with an ACK and another FIN and the connection is closed.

3. Connection Termination

Connection termination steps are

1. The client sends a FIN (finish) segment to notify the server that it no longer wants to send data. It sends its own sequence number, just as it does when the connection is established.
2. The server acknowledges receipt of the package with an ACK segment that contains the sequence number plus 1.
3. When the server has finished the data transfer, it also sends a FIN packet, to which it adds its sequence number.
4. Now it is the client's turn to send an ACK packet including the sequence number plus 1, which officially terminates the TCP connection for the server.


TCP connection termination (TCP Teardown)
Client
Server
FIN | SEQ. Client
ACK | SEQ. Client + 1
FIN | SEQ. Server
ACK | SEQ. Server + 1
SEQ. = Sequence number