NETWORK LAYER

Ques 1) What is network layer? What are the functions of network layer?

Or

List the network layer functions. (2019[03])

Or

Explain any two functions of network layer. (2020[1.5])

Ans: Network Layer

The network layer is the third layer of OSI model. It responds to service requests from the transport layer and issues service requests to the data link layer. Network layer addresses messages and translates logical addresses and names into physical addresses.

It also determines the route from the source to the destination computer and manages traffic problems, such as switching, routing, and controlling the congestion of data packets.

Functions of Network Layer

The specific functions of the network layer are given below:

1) **Logical Addressing:** The physical addressing implemented by the data link layer handles the addressing problem locally. If a packet passes the network boundary, we need another addressing system to help distinguish the source and destination systems.

The network layer adds a header to the packet coming from the upper layer that, among other things, includes the logical addresses of the sender and receiver.

2) **Routing:** When independent networks or links are connected together to create an Internetwork (a network of networks) or a large network, the connecting devices (called routers or gateways) route the packets to their final destination. One of the functions of the network layer is to provide this mechanism.

3) **Internetworking:** This is the main duty of network layer. It provides the logical connection between different types of networks.

4) **Packetising:** The network layer encapsulates the packets received from upper layer protocol and makes new packets. This is called as **packetising**. It is done by a network layer protocol called IP (Internetworking Protocol

5) **Fragmenting:** *The datagram can travel th[rough] different networks. Each router decapsulates t[he] datagram from the received frame. Then the data[gram] is processed and encapsulated in another frame.*

Ques 2) What are the different design issues [of] network layer?

Ans: Design Issues of Network Layer

The network layer design issues include:

1) **Services Provided to Transport Layer:** Main featur[es] of the services provided to transport layer are as follows:

   i) The services provided should be **independent o[f] the underlying technology**. Users of the service need not be aware of the physical implementation of the network – for all they know, their messages could be transported *via* carrier pigeon! This design goal has great importance when one consider the great variety of networks in operation.

   In the area of Public networks, networks in underdeveloped countries are nowhere near the technological ability of those in the countries like the US or Ireland. The design of the layer must not disable from connecting to networks of different technologies.

   ii) The **transport layer (i.e., the host computer) should be shielded from the number, type and different topologies of the subnets uses**. That is, all the transport layer wants is a communication link; it need not know how that link is made.

   iii) The **network addresses made available to the transport layer** should use a uniform numbering plan even across LANs and WANs.

2) **Internal Design of Subnet:** There are basically two different philosophies for organising the subnet:

   i) **Connections:** In the context of the internal operation of the subnet, a connection is usually called a **virtual circuit**.

   ii) **Connectionless:** The independent packets of the connectionless organisation are called **datagrams.**

Ques 3) What is difference between connection oriented and connectionless service?

Ans: Difference between Connection Oriented and Connectionless Service

Table below shows the difference between connection oriented and connectionless service:

Table 3.1: Connection-Oriented vs. Connection-Less Service

Basis	Connection-Oriented	Connection-Less
Connection	Prior connection needs to be established.	No prior connection is established.
Resource Allocation	Resources need to be allocated.	No prior allocation of resource is required.
Reliability	It ensures reliable transfer of data.	Reliability is not guaranteed as it is a best effort service.
Congestion	Congestion is not at all possible.	Congestion can occur likely.
Transfer mode	It can be implemented either using Circuit Switching or VCs.	It is implemented using Packet Switching.
Retransmission	It is possible to retransmit the lost data bits.	It is not possible.
Suitability	It is suitable for long and steady communication.	It is suitable for bursty transmissions.
Signalling	Connection is established through process of signalling.	There is no concept of signalling.
Packet Travel	In this packets travel to their destination node in a sequential manner.	In this packets reach the destination in a random manner.
Delay	There is more delay in transfer of information, but once connection established faster delivery.	There is no delay due absence of connection establishment phase.

ROUTING ALGORITHM

Ques 4) What is routing? What are the design goals of routing algorithm? List out the different types of routing algorithms.

Or

Define router and routing. (2020[1.5])

Ans: Routing

Routing is the process of selecting paths in a network along which to send network traffic. Routing is usually performed by a dedicated device called a router. A **router** is a networking device that forwards packets between networks using information in protocol headers and forwarding tables to determine the best next router for each packet. Routers work at the Network Layer (layer 3) of the OSI model and the Internet Layer of TCP/IP.

For routing of packets, routing algorithms are used. The routing algorithm is that part of the network layer software responsible for deciding which output line an incoming packet should be transmitted on.

Routing algorithms can be differentiated based on several key characteristics:

1) First, the particular goals of the algorithm designer affect the operation of the resulting routing protocol.
2) Second, various types of routing algorithms exist, and each algorithm has a different impact on network and router resources.
3) Finally, routing algorithms use a variety of metrics that affect calculation of optimal routes.

Design Goals of Routing Algorithms

1) **Optimality:** Optimality refers to the capability of the routing algorithm to select the best route, which depends on the metrics and metric weightings used to make the calculation.

For example, one routing algorithm may use a number of hops and delays, but it may weigh delay more heavily in the calculation. Naturally, routing protocols must define their metric calculation algorithms strictly.

2) **Simplicity and Low Overhead:** Routing algorithms also are designed to be as simple as possible. In other words, the routing algorithm must offer its functionality efficiently, with a minimum of software and utilisation overhead. Efficiency is particularly important when the software implementing the routing algorithm must run on a computer with limited physical resources.

3) **Robustness and Stability:** Routing algorithms must be robust, which means that they should perform correctly in the face of unusual or unforeseen circumstances, such as hardware failures, high load conditions, and incorrect implementations. Because routers are located at network junction points, they can cause considerable problems when they fail. The best routing algorithms are often those that have withstood the test of time and that have proven stable under a variety of network conditions.

4) **Rapid Convergence:** In addition, routing algorithms must converge rapidly. **Convergence** is the process of agreement, by all routers, on optimal routes. When a network event causes routes to either go down or become available, routers distribute routing update messages that permeate networks, stimulating recalculation of optimal routes and eventually causing all routers to agree on these routes. Routing algorithms that converge slowly can cause routing loops or network outages.

5) **Flexibility:** Routing algorithms should also be flexible, which means that they should quickly and accurately adapt to a variety of network circumstances. Assume,

for example, that a network segment has gone down. As many routing algorithms become aware of the problem, they will quickly select the next-best path for all routes normally using that segment. Routing algorithms can be programmed to adapt to changes in network bandwidth, router queue size, and network delay, among other variables.

Types of Routing Algorithm

Internet routing protocols employ one of following algorithms to gathering and using routing information:

1) Shortest Path Routing
2) Link-State Routing
3) Distance-Vector Routing
4) Flood-Based Routing Algorithm
5) Ad-Hoc On-Demand Distance Vector (AODV) Routing

Ques 5) Define the routing table.

Ans: Routing Table

Routing table is an electronic document that stores routes to various nodes in a computer network. The nodes may be any kind of electronic device connected to the network. The routing table is usually stored in a router or networked computer in the form of a database or file. When data needs to be sent from one node to another on the network, the routing table is referred to in order to find the best possible route for the transfer of information.

Network id	Cost	Next hop
....	....	....
....	....	....

Figure 3.1: Format of Routing Table

The routing table consists of at least three information fields:

1) **Network ID:** i.e., the destination network id.
2) **Cost:** i.e., the cost or metric of the path through which the packet is to be sent.
3) **Next Hop:** The next hop, or gateway, is the address of the next station to which the packet is to be sent on the way to its final **destination**.

Ques 6) Explain the Optimality Principle.

Ans: Optimality Principle

Optimality principle states that if router J is on the optimal path from router I to router K, then the optimal path from J to K also falls along the same route.

To see this, call the part of the route from I to J r_1 and the rest of the route r_2. If a route better than r_2 existed from J to K, it could be concatenated with r_1 to improve the route from I to K, contradicting our statement that r_1r_2 is optimal. As a direct consequence of the optimality principle, we can see that the set of optimal routes from all sources to a given destination form a tree rooted at the destination.

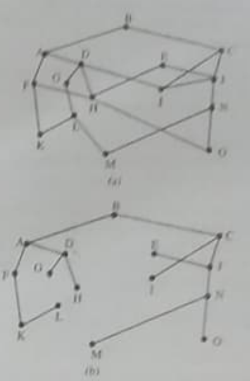


Figure 3.2: (a) A Subnet. (b) A Sink Tree for Router B

Such a tree is called a sink tree and is illustrated in **figure 3.2,** where the distance metric is the number of hop. Sink tree is not necessarily unique; other trees with the same path lengths may exist.

The goal of all routing algorithms is to discover and use the sink trees for all routers.

SHORTEST PATH ROUTING

Ques 7) Discuss the shortest path routing. Also explain the Dijkstra's Algorithm in detail.

Ans: Shortest Path Routing

Shortest Path Routing is suited for static routing. A path selected can be called shortest in many contexts. If one selects cost as criteria then the shortest path is the route which is least expensive. If distance is the criteria for determining shortest path then minimum length path is taken in to consideration.

If time is the criteria then the path which takes least time to reach the destination is called shortest path. One can use anyone of the following shortest path routing algorithms:

1) Dijkstra Algorithm
2) Bellman-Ford Algorithm

Dijkstra Algorithm

In this algorithm the criteria for shortest path is distance. All distances being known, in this method the shortest path with respect to distance is looked for from source to destination.

This is also called minimum cost or **Least cost algorithm.** The vertices are assumed to act as routers and edges act as connecting media. Dijkstra's algorithm is used to find the shortest path between any two vertices s and t in G.

The principle behind Dijkstra's algorithm is that if s, . . . x, . . . t is the shortest path from s to t, then s, . . . x had better be the shortest path from s to x.

This suggests a dynamic programming-like strategy, where one store the distance from s to all nearby nodes, and use them to find the shortest path to more distant nodes.

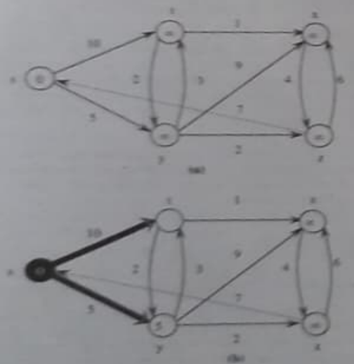
The shortest path from s to s, d(s, s) = 0. If all edge weights are positive, the smallest edge incident to s, say (s, x), defines d(s, x).

An array is used to store the length of the shortest path to each node. Initialize each to ∞ to start. Soon as the shortest path is established from s to a new node x, go through each of its incident edges to see if there is a better way from s to other nodes through x.

Steps: Dijkstra Algorithm

```
known = {s}
for i = 1 to n, dist[i] = ∞
for each edge (s, v), dist[v] = d(s, v)
last=s
while (last ≠ t)
select v such that dist(v) = min dist(i)
for each (v, x), dist[x] = min(dist[x], dist[v]+ w(v, x))
last = v
known = known U {v}
```

For example, consider **figure 3.3** in which source s is the leftmost vertex. The shortest-path estimates are shown within the vertices, and shaded edges indicate predecessor values. Black vertices are in the set S, and white vertices are in the min-priority queue Q = V – S.



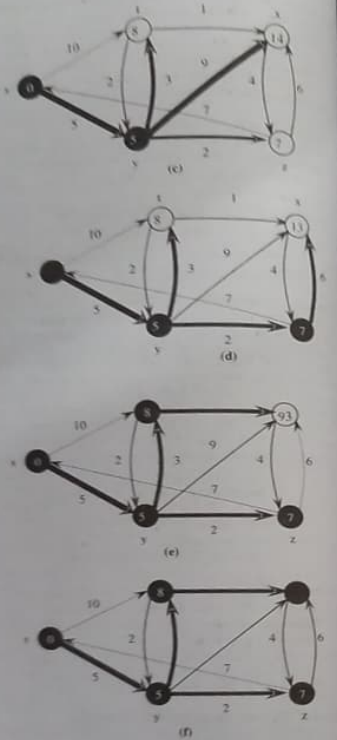


Figure 3.3: Execution of Dijkstra's Algorithm

In **Figure 3.3(a)** the situation just before the first iteration of the while loops of lines 4-8. The shaded vertex has the minimum d value and is chosen as vertex u in line 5. In **Figure 3.3(b)-(f)** the situations after each successive iteration of the while loop are shown. The shaded vertex in each part is chosen as vertex u in line 5 of the next iteration. The d value shown in part (f) is the final values. So the shortest path from s to t is 8, s to y is 5, s to z is 7, s to x is 9

Ques 8) Discuss the Bellman-Ford Algorithm with suitable example.

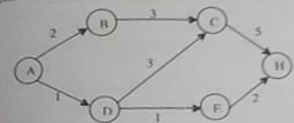
Ans: Bellman-Ford Algorithm

This algorithm is suitable for a directed graph. In this case to the least cost distance from every node in a network to a special node is found out. If for a given graph it is required to find the minimum path from all nodes to A then we proceed in such a way that we consider all those nodes which can reach that particular node in a single hop. Each node is marked in the format,

$$D_x^y = d$$

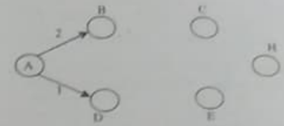
Where x is the node number, y the hops considered and d is the distance. If there are nodes which are not directly connected we mark their d as infinity. We continue evaluating this distance Vs hops till we have considered hop value upto one less than the number of nodes.

Example: Consider the following graph find the shortest path between node A and node H using Bellman-Ford algorithm.



Solution:

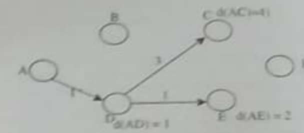
Step 1: Distance AD is shorter than AB. So route AD is chosen.



Step 2:

∵ d(AE) < d(AC)

∴ d(AE) is chosen.



Step 3: So the shortest distance is ADEH, the result is same as in Dijkstra's algorithm.

Ques 9) Differentiate between static and dynamic routing. (2018[03])

Ans: Difference between Static and Dynamic Routing

Basis for Comparison	Static Routing	Dynamic Routing
Configuration	Manual	Automatic
Routing table building	Routing locations are hand-typed	Locations are dynamically filled in the table.
Routes	User defined	Routes are updated according to change in topology.
Routing algorithms	Doesn't employ complex routing algorithms.	Uses complex routing algorithms to perform routing operations.
Implemented in	Small networks	Large networks
Link failure	Link failure obstructs the rerouting.	Link failure doesn't affect the rerouting.
Security	Provides high security.	Less secure due to sending broadcasts and multicasts.
Routing protocols	No routing protocols are indulged in the process.	Routing protocols such as RIP, EIGRP, etc are involved in the routing process.
Additional resources	Not required	Needs additional resources to store the information.

Ques 10) What do you understand by flooding?

Or

Describe the static routing algorithm flooding. (2020[03])

Or

What is flooding? Describe any two situations where flooding is advantageous. (2018[03])

Ans: Flood-based Routing Algorithm/Flooding

Flooding occurs when a router uses a non-adaptive routing algorithm to send an incoming packet to every outgoing link except the node on which the packet arrived. Flooding is a way to distribute routing protocols updates quickly to every node in a large network. Examples of these protocols include the **Open Shortest Path First** and **Distance Vector Multicast Routing Protocol**.

Flooding adapts the technique in which every incoming packet is sent on every outgoing line except the one on which it arrived. One problem with this method is that packets may go in a loop. As a result of this, a node may receive several copies of a particular packet which is undesirable.

Some techniques adapted to overcome these problems are as follows:

1) **Sequence Numbers:** Every packet is given a sequence number. When a node receives the packet it sees its source address and sequence number. If the node finds that it has sent the same packet earlier then it will not transmit the packet and will just discard it.

2) **Hop Count:** Every packet has a hop count associated with it. This is decremented (or incremented) by one by each node which sees it. When the hop count becomes zero (or a maximum possible value) the packet is dropped.

3) **Spanning Tree:** The packet is sent only on those links that lead to the destination by constructing a spanning tree routed at the source. This avoids loops in transmission but is possible only when all the intermediate nodes have knowledge of the network topology. Flooding is not practical for general kinds of applications. But in cases where high degree of robustness is desired such as in military applications, flooding is of great help. Flood-based routing, as the name suggests, uses redundant replication of incoming packets/NLDUs on available outgoing links.

Variants of Flood-based Routing

1) **Pure Flooding Algorithm:** This is one of the simplest algorithms available to date that has a simple logic that suggests that if a packet arrives at a node that is member of the flood-based routing architecture, simply copy it (by replicating the original) on all outgoing links other than the link going back to the node wherefrom the packet has just arrived.

   Although under extreme unpredictability, this algorithm demonstrates consistent robustness and guaranteed delivery as long as at least one path leading to the destination is available, it is inherently an inefficient algorithm due to the possibility of indefinite circulation of packets/NLDUs.

2) **Hop-Count based Flooding Algorithm:** This algorithm may be expressed as follows:

   i) At any originating node 's', structure a packet such that its header contains a 'hop-count' that be initialised to length of the path (if known) or full diameter of the subnet.

   ii) At every intermediate node 'i', examine the incoming queue of packets, take the packet at the head of the queue and note the packet-id, line on which it arrived on, its hop count and destination address.

   iii) Decrement the hop-count by one '1'.

   iv) If the count becomes zero, discard/drop the packet and flush the corresponding entries in the local table.

   v) Otherwise, generate (n – 1) replicas of the packet (where 'n' is the number of arcs converging at this node) and transmit one replica on each of the arcs/lines except the one this packet arrived on.

   vi) Examine the incoming queue and if it is non-empty, repeat steps 2-5 else wait until a new packet arrives and then repeat steps 2-5.

3) **Selective/Direction-constrained Flooding Algorithm:** It is a variant of the basic flooding algorithm with the constraint of direction thrown in for the purpose of improved efficiency. In this scheme, packets are selectively flooded by the routers in such a way that they move approximately in the right direction (i.e., leading towards the destination).

Situations where Flooding is Advantageous

Flooding is not practical in most applications, but it does have some uses:

1) In military applications, where large numbers of routers may be blown to bits at any instant, the tremendous robustness of flooding is highly desirable.

2) In distributed database applications, it is sometimes necessary to update all the databases concurrently, in which case flooding can be useful.

3) Another possible use of flooding is as a metric against which other routing algorithms can be compared. Flooding always chooses the shortest path, because it chooses every possible path in parallel. Consequently, no other algorithm can produce a shorter delay (if we ignore the overhead generated by the flooding process itself).

Ques 11) Differentiate between Flooding and Broadcasting. (2019[03])

Ans: Difference between Flooding and Broadcasting

Table 3.2 shows the difference between Flooding and Broadcasting:

Table 3.2: Difference between Flooding and Broadcasting

Flooding	Broadcasting
Flooding is a very simple routing algorithm which sends all incoming packets through every outgoing edge.	Broadcasting is a method used in computer networking, which makes sure that every device in the network will receive a (broadcasted) packet.
Flooding does not send packets to all hosts simultaneously. The packets would ultimately reach all nodes in the network due to flooding.	Sending a packet to all hosts simultaneously is broadcasting.
Flooding may send the same packet along the same link multiple times.	Broadcasting sends a packet along a link at most once.
Several copies of the same packet may reach nodes in flooding.	Broadcasting does not cause that problem. Unlike flooding, broadcasting is done by specifying a special broadcast address on packets.

Ques 12) Describe the Distance Vector Routing Algorithm in detail.

Or

Discuss the problems occurred in distance vector routing.

Or

Explain/Illustrate distance vector routing with an example. (2018[06]-2020[05])

Ans: Distance Vector Routing Algorithm

Distance Vector Routing (DVR) is also known as the **Bellman-Ford or Ford-Fulkerson routing algorithm.** It is the original dynamic routing algorithm used in the erstwhile ARPANET.

This scheme may be expressed as:

1) Each router knows/discovers its distance from its neighbours.

2) Each router locally maintains a routing table indexed by an entry for every other router in the subnet and identification of a preferred neighbour/link leading to that router.

3) Metric of estimation may vary. **For example,** it may be any one of physical distance, hops, delay, etc.,

4) Periodically, each router sends a vector to its neighbouring routers. As this vector contains estimated distances, it is called a distance vector.

5) On receipt of such vectors from its neighbours, every router revises its estimates and updates its local routing table.

Problems in Distance Vector Routing

The problem which arises in the Distance Vector Algorithms is given below:

1) **Count-to-Infinity Problem:** Consider a router whose best route to destination X is large. If on the next exchange neighbour A suddenly reports a short delay to X, the router just switches over to using the line to A to send traffic to X. In one vector exchange, the good news is processed. **For example,** let us consider the five-node (linear) subnet of **figure 3.4**, where the delay metric is the number of hops. Suppose A is down initially and all the other routers know this. In other words, they have all recorded the delay to A as infinity.

2) When A comes up, the other routers learn about it via the vector exchanges. For simplicity assume that there is (very big warning bell) somewhere that is struck periodically to initiate a vector exchange at all routers simultaneously. At the time of the first exchange, B learns that its left neighbour has zero delay to A. B now makes an entry in its routing table that A is one hop away to the left. All the other routers still think that A is down. At this point, the routing table entries for a are as shown in the second row of the **figure 3.4 (a)**.

3) When A comes up, the other routers learn about it via the vector exchanges. For simplicity assume that there is (very big warning bell) somewhere that is struck periodically to initiate a vector exchange at all routers simultaneously.

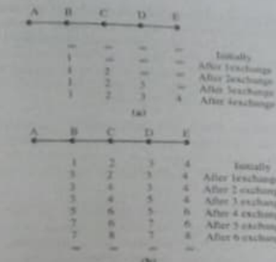


Figure 3.4: Count-to-infinity Problem

At the time of the first exchange, B learns that its left neighbour has zero delay to A. B now makes an entry in its routing table that A is one hop away to the left. All the other routers still think that A is down. At this point, the routing table entries for a are as shown in the second row of the **figure 3.4 (a)**.

On the next exchange, C learns that B has a path of length 1 to A, so it updates its routing table to indicate a path of length 2, but D and E do not hear the good news until later.

Clearly, the good news is spreading at the rate of one hop per exchange. In a subnet whose longest path is of length N hops, within N exchanges everyone will know about newly revived lines and routers.

Consider the situation of **figure 3.4 (b)** in which all the lines and routers are initially up. Routers B, C, D, and E have distances to A of 1, 2, 3, and 4, respectively. Suddenly A goes down, or alternatively, the line between A and B is cut, which is effectively the same thing from B's point of view.

At the first packet exchange, B does not hear anything from A. Fortunately, C says "Do not worry. I have a path to A of length 2." Little does B know that C's path runs through B itself? For all B knows, C might have ten outgoing lines all with independent paths to A of length 2. As a result, B now thinks it can reach A via C, with a path length of 3. D and E do not update their entries for A on the first exchange. On the second exchange, C notices that each of its neighbours claims to have a path to A of length 3. It picks one of them at random and makes its new distance to A 4, as shown in the third row of **figure 3.4 (b)**. Subsequent exchanges produce the history shown in the rest of **figure 3.4 (b)**.

From this figure, it is clear no router ever has a value more than one higher than the minimum of all its neighbours. Gradually, all the routers work their way to infinity, but the number of exchanges required depends on the numerical value used for infinity. For this reason, it is wise to set infinity to the longest path plus 1. If the metric is time delay, there is no well-defined upper bound, so a high value is needed to prevent a path with a long delay from being treated as down. This problem is known as the count-to-infinity problem.

4) **Split Horizon Hack:** The split horizon algorithm works the same way as distance vector routing, except that the distance to X is not reported on the line that packets for X are sent on (actually, it is reported as infinity).

In the initial state of above **figure 3.5 (b), for example,** C tells D the truth about the distance to A, but C tells B that its distance to A is infinite. Similarly, D tells the truth to E but lies to C.

On the first exchange, B discovers that the direct line is gone, and C is reporting an infinite distance to A as well. Since neither of its neighbours can get to A, B sets its distance to infinity as well. On the next exchange, C hears that A is unreachable from both of its neighbours, so it marks A as unreachable too.

Using split horizon, the bad news propagates one hop per exchange. This rate is much better than without split horizon. The split horizon, although widely used, sometimes fails.

Consider, **for example,** the four-node subnet of **Figure 3.5**. Initially, both A and B have a distance 2 to D, and C has a distance 1 there. Now suppose that the CD line goes down. Using split horizon, both A and B tell C that they cannot get to D. Thus C immediately concludes that D is unreachable and reports this to both A and B. Unfortunately, A hears that B has a path of length 2 to D, so it assumes it can get to D via B in 3 hops. Similarly, B concludes it can get to D via A in 3 hops. On the next exchange, they each set their distance to D to 4. Both of them gradually count to infinity, precisely the behaviour we were trying to avoid.

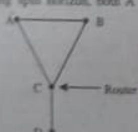


Figure 3.5: An Example where Split Horizon Fails

Ques 13) Explain link state routing algorithm.

Or

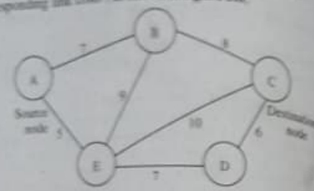
Explain the different steps in link state routing. (2018[05])

Or

Explain how routing is performed using link state algorithm? Illustrate with an example. (2019[06])

Ans: Link State Routing Algorithm

In this algorithm, exchange of the link state packets over the subnet hold key to facilitating the routing process. In this algorithm, network topology and link costs are estimated by making each node broadcast what is referred to as 'link state packets' carrying 'identities of neighbours' and 'corresponding link costs', as shown in **figure 3.6**.



A	
11...001	
60	
B	7
E	5

Destination (router)	Link-cost	Next hop (router)	Hop count
A	0	A	1
B	7	B	1
C	15	B	2
D	12	E	2
E	5	E	1

Source	Sequence No.	Age	Send Flags	Acknowledgement Flags	Data

Figure 3.6: Structure of a Link-State Packet, Routing Table and Packet Buffer (at router A)

The basic idea involves the computation of the local routing table by each router on the basis of its own estimates and the similar link-state broadcasts received from other routers in the subnet.

In a simple version each router following this algorithm:

Step 1) Discovers its neighbours and their network addresses by sending special packets called 'hello' packets.

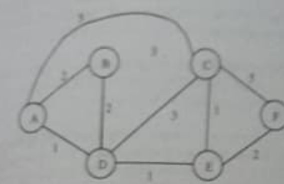
Step 2) Estimates delay/cost or any other metric for reaching its neighbours by sending another special packet called 'echo' packets.

Step 3) Immediately applies its recent knowledge to form link-state packet, which encapsulates this estimate; and, sends (broadcasts) the packet to all the discovered routers.

Step 4) Computes the shortest path to every other router using the shortest path algorithm and updates the local routing table.

Step 5) Immediately forms fresh Link-State Packets (LSPs) and executes link-state broadcast. This is sometimes called **controlled flooding**.

Example 1: Let us consider the following example:



In the **above figure**, source vertex is A.

Step 1: The first step is an initialization step. The currently known least cost path from A to its directly attached neighbors, B, C, D are 2,5,1 respectively. The cost from A to B is set to 2, from A to D is set to 1 and from A to C is set to 5. The cost from A to E and F are set to infinity as they are not directly linked to A.

Step	N	D(B),P(B)	D(C),P(C)	D(D),P(D)	D(E),P(E)	D(F),P(F)
1	A	2,A	5,A	1,A	∞	∞

Step 2: In the above table, we observe that vertex D contains the least cost path in step 1. Therefore, it is added in N. Now, we need to determine a least-cost path through D vertex.

i) Calculating shortest path from A to B

v = B, w = D
D(B) = min(D(B) , D(D) + c(D,B))
= min(2, 1+2)
= min(2, 3)

The minimum value is 2. Therefore, the currently shortest path from A to B is 2.

ii) Calculating shortest path from A to C

v = C, w = D
D(B) = min(D(C) , D(D) + c(D,C))
= min(5, 1+3)
= min(5, 4)

The minimum value is 4. Therefore, the currently shortest path from A to C is 4.

iii) Calculating shortest path from A to E

v = E, w = D
D(B) = min(D(E) , D(D) + c(D,E))
= min(∞, 1+1)
= min(∞, 2)

The minimum value is 2. Therefore, the currently shortest path from A to E is 2.

Step	N	D(B),P(B)	D(C),P(C)	D(D),P(D)	D(E),P(E)	D(F),P(F)
1	A	2,A	5,A	1,A	∞	∞
2	AD	2,A	4,D		2,D	∞

Step 3: In the above table, we observe that both E and B have the least cost path in step 2. Let's consider the E vertex. Now, we determine the least cost path of remaining vertices through E.

i) Calculating the shortest path from A to B

v = B, w = E
D(B) = min(D(B) , D(E) + c(E,B))
= min(2 , 2+ ∞)
= min(2, ∞)

The minimum value is 2. Therefore, the currently shortest path from A to B is 2.

ii) Calculating the shortest path from A to C.

v = C, w = E
D(B) = min(D(C) , D(E) + c(E,C))
= min(4 , 2+1)
= min(4,3)

The minimum value is 3. Therefore, the currently shortest path from A to C is 3.

iii) Calculating the shortest path from A to F.

v = F, w = E
D(B) = min(D(F) , D(E) + c(E,F))
= min(∞ , 2+2)
= min(∞ ,4)

The minimum value is 4. Therefore, the currently shortest path from A to F is 4.

Step	N	D(B),P(B)	D(C),P(C)	D(D),P(D)	D(E),P(E)	D(F),P(F)
1	A	2,A	5,A	1,A	∞	∞
2	AD	2,A	4,D		2,D	∞
3	ADE	2,A	3,E			4,E

Step 4: In the above table, we observe that B vertex has the least cost path in step 3. Therefore, it is added in N. Now, we determine the least cost path of remaining vertices through B.

i) Calculating the shortest path from A to C.

v = C, w = B
D(B) = min(D(C) , D(B) + c(B,C))
= min(3 , 2+3)
= min(3,5)

The minimum value is 3. Therefore, the currently shortest path from A to C is 3.

ii) Calculating the shortest path from A to F.

v = F, w = B
D(B) = min(D(F) , D(B) + c(B,F))
= min(4, ∞)
= min(4, ∞)

The minimum value is 4. Therefore, the currently shortest path from A to F is 4.

Step	N	D(B),P(B)	D(C),P(C)	D(D),P(D)	D(E),P(E)	D(F),P(F)
1	A	2,A	5,A	1,A	∞	∞
2	AD	2,A	4,D		2,D	∞
3	ADE	2,A	3,E			4,E
4	ADEB		3,E			4,E

Step 5: In the above table, we observe that C vertex has the least cost path in step 4. Therefore, it is added in N. Now, we determine the least cost path of remaining vertices through C.

Calculating the shortest path from A to F:

v = F, w = C

D(B) = min(D(F) , D(C) + c(C,F))

= min(4, 3+5)

= min(4,8)

The minimum value is 4. Therefore, the currently shortest path from A to F is 4.

Step	N	D(B),P(B)	D(C),P(C)	D(D),P(D)	D(E),P(E)	D(F),P(F)
1	A	2,A	5,A	1,A	∞	∞
2	AD	2,A	4,D		2,D	∞
3	ADE	2,A	3,E			4,E
4	ADEB		3,E			4,E
5	ADEBC					4,E

Final table is shown below:

Step	N	D(B),P(B)	D(C),P(C)	D(D),P(D)	D(E),P(E)	D(F),P(F)
1	A	2,A	5,A	1,A	∞	∞
2	AD	2,A	4,D		2,D	∞
3	ADE	2,A	3,E			4,E
4	ADEB		3,E			4,E
5	ADEBC					4,E
6	ADEBCF					

Ques 14) Give the relevance of age field in a link state packet. (2019[03])

Ans: Link State Algorithm has a few problems, but they are manageable:

1) First, if the sequence numbers wrap around, confusion will reign. The solution here is to use a 32-bit sequence number. With one link state packet per second, it would take 137 years to wrap around, so this possibility can be ignored.

2) Second if a router ever crashes, it will lose track of its sequence number. If it starts again at 0, the next packet will be rejected as a duplicate.

3) Third, if a sequence number is ever corrupted and 65,540 is received instead of 4(a 1-bit error), packet 5 through 65,540 will be rejected as obsolete, since the current sequence number is thought to be 65,540.

The solution of all these problems is to include the age of each packet after the sequence number and decrement it once per second. When the age hits zero, the information from that router is discarded. Normally, a new packet comes in, say, every 10 sec, so router information only times out when a router is down(or six consecutive packets have been lost, an unlikely event). The **Age field** is also decremented by each router during the initial flooding process, to make sure no packet can get lost and live for an indefinite period of time (a packet whose age is zero is discarded).

Ques 15) Explain the multicast routing in detail.

Ans: Multicast Routing

To send messages to well defined groups that are numerically large in size but small compared to the network as a whole. Sending message to such a group is called **multicasting** and its routing algorithm is called **multicasting routing**.

Multicasting requires group management. Some way is needed to create and destroy groups and to allow processes to join and leave groups.

To do multicasting routing each router computes a spanning tree covering all other routers. **For example,** in **figure 3.7(a)** we have a subnet with two groups, 1 and 2. Some routers are attached to hosts that belong to one or both of these groups, as indicated in the figure. A spanning tree for the leftmost router is shown in **figure 3.7(b)**.

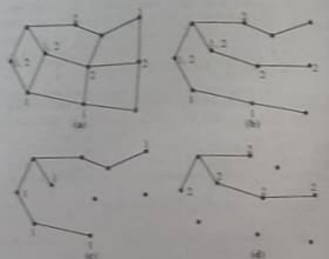


Figure 3.7: (a) A Subnet (b) A spanning tree for the leftmost router.
(c) A multicast tree for a group 1. (d) A multicast tree for group 2

When a process sends a multicast packet to a group, the first router examines its spanning tree and prunes it, removing all lines that do not lead to hosts that are members of the group. In our example, **figure 3.7(c)** shows the pruned spanning tree for group1.

Similarly, **figure 3.7(d)** shows the pruned spanning tree for group 2. Multicast packets are forwarded only along the appropriate spanning tree. No hosts interested in a particular group and not connection to other routers receives a multicast message for that group, it responds with a PRUNE message, telling the sender not to send it any more multicasts for that group.

When a router with no group members among its own hosts has received such messages on all its lines, it, too, can respond with a PRUNE message. In this way, the subnet is recursively pruned.

One **potential disadvantage** of this algorithm is that it scales poorly to large network. Support that a network has n groups, each with an average of m members.

For each group m pruned spanning trees must be stored, for a total of mn trees. When many large groups exist, considerable storage is needed to store all the trees.

Ques 16) Explain the Routing for Mobile Hosts.

Or

What is mobile routing in the telephone network?

Or

Discuss about the routing for mobile hosts. (2019[04])

Or

Explain how to perform for mobile hosts.

Ans: Routing for Mobile Hosts

What happens if a destination is not attached by a wire to a router, but instead can move about? Packets destined to that host somehow have to be forwarded to its new location, wherever it may be. The problem naturally resolves itself into two parts:

1) Finding-out where a host is, and
2) Getting packets or calls to it.

Mobile Routing in the Telephone Network

Cellular telephones use radio frequencies to communicate with a base station – usually located on a tall tower with a triangular platform on top, which you can see along major highways or in city centers – that relays their call to a Mobile Telephone Switching Office (MTSO). (To prevent unfair advantage to the local telephone company, the Federal Communications Commission in the United States requires that MTSOs be separated from central offices, though they serve nearly the same purpose.) Routing calls to and from a cellular telephone that may be associated with any MTSO in the cellular-service provider's service area.

Each cellular phone is statically assigned a globally unique ID and a home MTSO that does billing and provides access to the long-distance (toll) telephone network. The phone is also assigned a telephone number from the address space assigned to the home MTSO. When a cellular phone is switched on, it uses ALOHA contention on a common signaling channel to identify itself to the local MTSO. The MTSO, in turn, contacts the home MTSO and informs it of the phone's location.

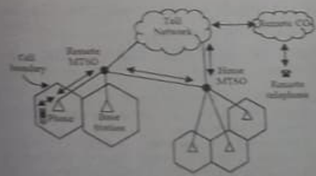


Figure 3.8: Routing for Cellular Phones

When someone makes a call to the phone, the telephone network delivers it automatically to the home MTSO, which sets-up a connection to the phone through the remote MTSO, using Signaling System 7(SS7) signaling (**figure 3.8**). The remote MTSO contacts the nearest base station, which rings the cellular phone. The identity of the nearest base station is dynamically updated using the cellular hand-off.

In the **figure 3.10**, Each Cellular Phone is assigned to a Mobile Telephone Switching Office (MTSO).

Calls to the Phone are Routed through the Home MTSO to the Nearest Base Station via a Remote MTSO. As the Phone Moves, the Home MTSO is updated with the Location of the MTSO nearest the Phone.

To keep billing and accounting simple, all calls from the phone are always routed back to the home MTSO before they enter the toll network. Thus, the remote MTSO acts like a dumb switch to route calls to and from the home MTSO. If the phone moves from one MTSO to another, this information is sent back to the home MTSO, which updates its local database. Calls in progress are re-routed from the remote MTSO to the home MTSO, again using SS7 signaling.

This architecture allows cellular phones to roam within the entire service area freely, but has the overhead that calls are always routed to the home MTSO, requiring additional hops in the network.

Ques 17) What is mobile Routing in the Internet?

Ans: Mobile Routing in the Internet

Extensions to the standard solution that add robustness, efficiency, and security are still areas of active research. The field has evolved its own set of acronyms, which are presented in **table 3.3**:

Table 3.3: Acronyms Used in Mobile Routing on the Internet

Acronym	Expansion	Comment
MH	Mobile Host	The host that moves.
CH	Corresponding Host	The host that the mobile is talking to.
HAA	Home Address Agent	The "home" base assigned to the mobile host.
COA	Care-of Agent	The base closest to the mobile host that forwards packets to it.

The basic model for mobile routing, which is similar to the cellular telephone model, is shown in **figure 3.9**. Mobile Hosts (MHs), which are mobile computers with a fixed IP address (much like a cellular phone with a fixed telephone number) communicate with the nearest base station, which is attached to a Care-of Agent (COA).

The care-of agent, which corresponds to a remote MTSO in cellular telephony, receives messages on behalf of the MH. We statically assign each MH to a Home Address Agent (HAA), which corresponds to a local MTSO. We call the machine that the MH is communicating with the corresponding host, or CH.

When a corresponding host wants to send a datagram to a mobile host, it puts the mobile host's IP address in the packet destination and hands it to the wide-area network. Using normal network routing, this packet eventually reaches the home address agent. The home address agent is always kept informed of the current care-of agent.

It encapsulates the incoming packet with a new header that shortly see how this is done). It encapsulates the incoming packet with a new header that has its destination set to that of the care-of agent (this is identical to the tunneling used in the MBONE). The care-of agent retrieves the packet and hands it to the base station, which sends it through a wireless link to the mobile host. When the mobile host wants to send a datagram to the corresponding host, it simply puts the corresponding host's IP address in the packet destination, and it is delivered to the corresponding host using normal routing.

This solution is nearly identical to the cellular network solution, except that we gain some efficiency in the path from the mobile host to the corresponding host, which does not need to go through the home address agent.

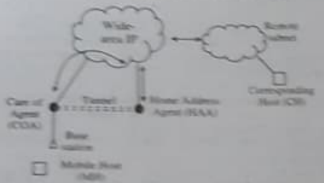


Figure 3.9: Mobile Routing on the Internet

In the figure 3.9, packets to a Mobile Host (MH) are always Routed through a Home Address Agent (HAA), which Tunnels Packets for the MH to a Care-of Agent (COA). When the MH Moves, it listens to Beacons to Detect that it has a New COA. It then Updates both the HAA and the Old COA. Packets from the MH use Normal Routing.

CONGESTION CONTROL

Ques 18) What is congestion? What are the causes of congestion?

Or

Show the effect on congestion on throughput of a network using a diagram.

Ans: Congestion

Congestion occurs in a computer network when the resource demands exceed the capacity. Packets may be lost due to too much queuing in the network. During congestion, the network throughput may drop and the path delay may become very high. Congestion in a network may occur if users send data into the network at a rate greater than that allowed by network resources.

For example, congestion may occur because the switches in a network have a limited buffer size to store arrived packets before processing. A congestion control scheme helps the network to recover from the congestion state. A congestion avoidance scheme allows a network to operate in the region of low delay and high throughput. Such schemes prevent a network from entering the congested state.

Causes of Congestion

The main causes of congestion over network are as below:
1) Unpredictable statistical fluctuation of traffic flows,
2) Fault conditions within the network, and
3) Slow processor speed. If the router's CPU speed is low and performing tasks like queuing buffers, table updating etc. queues are built up, even though the line capacity is not fully utilized.
4) Inefficient control policies (buffers not allocated fairly or correctly).
5) Bandwidth of the links is important in congestion. The links to be used must be of high bandwidth to avoid the congestion.

The effect on congestion on throughput of a network is shown in figure 3.10.

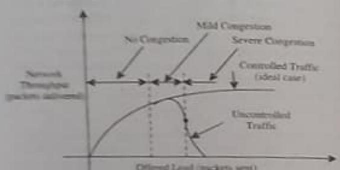


Figure 3.10: Effect of Congestion

Ques 19) What are the different congestion control methods?

Or

What is open-loop and closed-loop congestion control?

Or

List and explain any three closed loop congestion control techniques. (2018[03])

Or

Explain any three closed loop congestion control techniques. (2020[03])

Ans: Congestion Control Methods

The solutions to congestion problems can be divided into two categories or groups as:

1) **Open-Loop Congestion Control:** Open-loop congestion control algorithms which do not depend on any sort of direct feedback from the network. They can be implemented using a combination of CAC (Connection Admission Control) and UPC (Usage Parameter Control) procedures.

   The techniques used in open-loop congestion control are given below:

   i) **Retransmission Policy:** Retransmission is sometimes unavoidable. If the sender feels that a sent packet is lost or corrupted, the packet needs to be retransmitted.

   Retransmission in general may increase congestion in the network. However, a good retransmission policy can prevent congestion.

   ii) **Window Policy:** The type of window at the sender may also affect congestion. The Selective Repeat Window is better than the Go-Back-N window for congestion control. In the Go-Back-N window, when the timer for a packet times out, several packets may be resent, although some may have arrived safe and sound at the receiver.

   iii) **Acknowledgement Policy:** The acknowledgement policy imposed by the receiver may also affect congestion. If the receiver does not acknowledge every packet it receives, it may slow down the sender and help prevent congestion. Several approaches are used in this case. A receiver may send an acknowledgment only if it has a packet to be sent or a special timer expires.

   iv) **Discarding Policy:** A good discarding policy by the routers may prevent congestion and at the same time may not harm the integrity of the transmission.

   v) **Admission Policy:** An admission policy, which is a quality-of-service mechanism, can also prevent congestion in virtual-circuit networks. Switches in a flow check the resource requirement of a flow before admitting it to the network.

2) **Closed-Loop Congestion Control:** Closed loop congestion control algorithms adopt a method where the source recognizes network congestion by means of feedback information from the network. The source then limits the number of cells injected into the network by some appropriate method. The techniques used in closed-loop congestion control are given below:

   i) **Backpressure:** The technique of backpressure refers to a congestion control mechanism in which a congested node stops receiving data from the immediate upstream node or nodes. This may cause the upstream node to become congested and they, in turn, reject data from their upstream nodes or nodes.

   ii) **Choke Packet:** A choke packet is a packet sent by a node to the source to inform it of congestion. In backpressure, the warning is from one node to its upstream node, although the warning may eventually reach the source station.

   In the choke packet method, the warning is from the router, which has encountered congestion, to the source station directly. The intermediate nodes through which the packet has travelled are not warned.

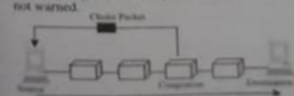


Figure 3.11: Choke Packet

   iii) **Implicit Signalling:** In implicit signalling, there is no communication between the congested node or nodes and the source. The source guesses that there is congestion somewhere in the network from other symptoms.

   iv) **Explicit Signalling:** The node that experiences congestion can explicitly send a signal to the source or destination. The explicit signalling method, however, is different from the choke packet method. In the choke packet method, a separate packet is used for this purpose; in the explicit signalling method, the signal is included in the packets that carry data. It is of two types:

   a) **Backward Signaling:** A bit can be set in a packet moving in the direction opposite to the congestion. This bit can warn the source that there is congestion and that it needs to slow down to avoid the discarding of packets.

   b) **Forward Signaling:** A bit can be set in a packet moving in the direction of the congestion. This bit can warn the destination that there is congestion. The receiver in this case can use policies, such as slowing down the acknowledgements, to alleviate the congestion.

Ques 20) List and explain various congestion control algorithms.

Or

Define leaky bucket and token bucket algorithm for congestion control.

Or

How token bucket algorithm performs congestion control? (2019[03])

Or

Explain any two congestion control algorithms. (2019[05])

Or

Demonstrate token bucket algorithm with a diagram. (2020[06])

Or

Explain the load shedding algorithm in detail.

Ans: Types of Congestion Control Algorithms

Congestion in a frame relay network is a problem that must be avoided because it decreases throughput and increases delay. Following are the three types of algorithm for congestion control:

1) **Leaky Bucket Algorithm:** If there is a hole at the bottom of a bucket, then no matter at what rate the bucket is filled up, the water leaks out drop by drop at a constant rate from the hole. Each host is connected by an interface that has finite queue acting like a "leaky bucket".

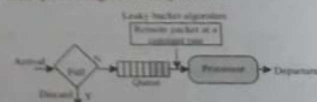


Figure 3.12: Leaky Bucket Implementation

When a packet comes to a host with the queue full, it is discarded. The host is allowed to put one packet per clock tick into the network. This can be enforced by the interface card or by the operating system. This converts an uneven flow of packets from the user process in an even flow of packets onto the network. Conceptually, each host is connected to the network by an interface containing a leaky bucket, that is, a finite internal queue. If a packet arrives at the queue when it is full, the packet is discarded.

Implementing the original leaky bucket algorithm is easy. The leaky bucket consists of a finite queue. When a packet arrives, if there is room on the queue it is appended to the queue; otherwise, it is discarded. At every clock tick, one packet is transmitted (unless the queue is empty).

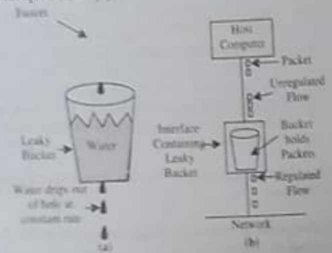


Figure 3.13: (a) A Leaky Bucket with Water and (b) A Leaky Bucket with Packets

2) **Token Bucket Algorithm:** The leaky bucket algorithm enforces a rigid output pattern at the average rate, no matter how burst the traffic is. For many applications, it is better to allow the output to speed up somewhat when large bursts arrive, so a more flexible algorithm is needed, preferably one that never loses data. One such algorithm is the **token bucket algorithm**. In this algorithm, the leaky bucket holds tokens, generated by a clock at the rate of one token every ΔT sec. For a packet to be transmitted, it must capture and destroy one token.

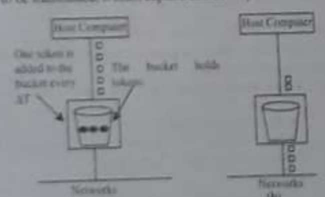


Figure 3.14: Token Bucket Algorithm (a) Before, (b) After

The leaky bucket algorithm does not allow idle hosts to save up permission to send large bursts later. The token bucket algorithm does allow saving, up to the maximum size of the bucket, n.

This property means that bursts of up to n packets can be sent at once, allowing some burstiness in the output stream and giving faster response to sudden burst of input. Another difference between the two algorithms is that the token bucket algorithm throws away tokens when the bucket fills up but never discards packets. In contrast, the leaky bucket algorithm discards packets when the bucket fills up.

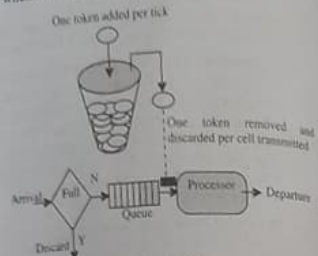


Figure 3.15: Token Bucket

If one call the burst length S sec, the token bucket capacity C bytes, the token arrival rate ρ bytes/sec, and the maximum output rate M bytes/sec, we see that an output burst contains a maximum of $C + \rho S$ bytes. We also know that the number of bytes in a maximum-speed burst of length S seconds in MS. Hence we have

$$C + \rho S = MS$$

We can solve this equation to get $S = C/(M - \rho)$.

3) **Load Shedding Algorithm:** Load shedding is the process of systematically reducing the system demand by temporarily decreasing the load in response to transmission or capacity shortages. Sometimes there simply may be too much traffic to be able to get it all through. When this happens, some packets must be lost. The packets are lost forever if the stream was unacknowledged; however, if the stream has some form of control, a retransmission can be tried at a later time. A router needs to decide how to choose which packets to drop. If the router knows something about the traffic, it might be possible to make intelligent choices; otherwise, packets are picked at random.

Load shedding is usually a last-ditch effort by routers when other congestion control methods are not alleviating the congestion problem. Load shedding simply means that the routers will dump packets they cannot process.

However, routers can be selective in which packets they discard instead of just dropping packets at random. In some types of applications, such as an FTP service, an old packet is more valuable than a new one.

Which packet to discard depends on the applications running there are two **policies**:

i) **Wine Policy:** For file transfer, an old packet is worth more than a new one. This is because dropping an old packet may force more packets to be re-transmitted (since receiver will discard out-of-order packets). For this kind of applications, "the older the better".

ii) **Milk Policy:** For multimedia, a new packet is more important than an old one. Thus, "fresher is better".

Implementing some sort of intelligent discard policy – For some applications, some packets are more important than others e.g., in MPEG video standard, periodically, an entire frame is transmitted and this is followed by subsequent frames as differences from the full reference frame → drop packets that is part of a difference is preferred to drop one that contains part of the last full reference frame.

Applications mark their packets in priority classes to indicate how important they are, such as very important – never discard, or lower priority, etc.

Ques 21) A host sends to a network via a token bucket. The token bucket has a capacity of 15 M bits and is filled with token at the rate of 5 M bits/second. Data is buffered if it arrives at the token bucket when there are no tokens. How long does it take for 30 M bits to enter the network assuming that the host sends at a peak rate of 20 M bits/second and the token bucket is initially full?

Ans: According to question

Token bus capacity = 15 M bits and token rate of 5 Mbps.

S = ?

Given, C = 30 M bits, M = 20 M bits/second, ρ = 5 M bits/sec.

Thus we know that,

$$S = \frac{C}{M - \rho} = \frac{30}{(20 - 5)} = \frac{30}{15} = 2 \text{ second}$$

Ques 22) What is difference between Token-bucket and leaky-bucket algorithm?

Ans: Difference between Token-Bucket and Leaky-Bucket algorithm

Table 3.4: Difference between Token-Bucket and Leaky-Bucket Algorithm

Token Bucket Algorithm	Leaky Bucket Algorithm
Token dependent	Token independent
If bucket is full token are discarded, but not the packet	If bucket is full packet or data is discarded
Packets can only transmit when there are enough token.	Packets are transmitted continuously.
It allows large bursts to be sent at faster rate after that constant rate.	It sends the packet at constant rate.
It saves token to send large bursts.	It does not save token.

QUALITY OF SERVICE (QOS)

Ques 23) What is meant by term QoS? What are the different flow characteristics?

Or

What is QoS? **(2019[02])**

Or

Write notes on QoS in networks. **(2020[03])**

Ans: Quality of Service

Quality of service is defined as something a flow seeks to attain. A stream of packets from a source to destination is called **flow**. In a connection-oriented network all packets belonging to a flow, follow the same route; in a connection-less service they may follow different routes.

The needs of each flow can be characterised by primary parameters viz. reliability, delay, jitter and bandwidth. Together these determine the QoS (Quality of Service) the flow requires.

QoS defines a set of attributes related to the performance of the connection. For each connection, the user can request a particular attribute.

Flow Characteristics

Traditionally, four types of characteristics are attributed to a flow which is given below:

1) **Reliability:** Reliability is a characteristic that a flow needs. Lack of reliability means losing a packet or acknowledgement, which entails retransmission. However, the sensitivity of application programs to reliability is not the same.

2) **Delay:** Source-to-destination delay is another flow characteristic. Again applications can tolerate delay in different degrees. In this case, telephony, audio conferencing, video conferencing and remote log-in need minimum delay, while delay in file transfer or e-mail is less important.

3) **Jitter:** Jitter is the variation in delay for packets belonging to the same flow. **For example,** if four packets depart at times 0, 1, 2 and 3 and arrive at 20, 21, 22 and 23, all have the same delay, 20 units of time.

Jitter is defined as the variation in the packet delay. High jitter means the difference between delays is large; low jitter means the variation is small.

4) **Bandwidth:** Different applications need different bandwidths. **In video conferencing** one need to send millions of bits per second to refresh a colour screen while the total number of bits in an e-mail may not reach even a million.

Ques 24) What are the different QoS attributes?

Ans: QoS Attributes

The attributes can be classified into two major categories as below:

1) **User-Related Attributes:** These attributes are related to the end user in the sense that they define how fast a user wants to send/receive data. These attributes are negotiated and defined at the time of the contract between the user and the network service provider. **Table 4.2** summarises user-related attributes.

Table 3.5: User-Related QoS Attributes

Attribute	Description
Sustained Cell Rate (SCR)	This is the average cell rate over a period of time, which could be more or less the actual transmission rates, as long as the average is maintained.
Peak Cell Rate (PCR)	This is the maximum transmission rate at a point of time.
Minimum Cell Rate (MCR)	This is the minimum cell rate that the network guarantees a user.
Cell Variation Delay Tolerance (CVDT)	This is a unit of measuring the changes in cell transmission times (i.e., what is the maximum and minimum delay between the delivery of any two cells).

2) **Network-Related Attributes:** These attributes define the characteristics of a network.

Table 4.3 summarises network-related attributes.

Table 3.6: Network-Related QoS Attributes

Attribute	Description
Cell Loss Ratio (CLR)	This attribute defines the fraction of the cells lost/delivered too late during transmission.
Cell Transfer Delay (CTD)	This is the average time required for a cell to travel from the source to the destination.
Cell Delay Variation (CDV)	This is the difference between the maximum and minimum values of CTD.
Cell Error Ratio (CER)	This parameter defines the fraction of cells that contained errors.

Ques 25) Discuss the requirements of Quality of Service (QoS).

Ans: Requirements of Quality of Service (QoS)

Quality of service (QoS) requirements are technical specifications that specify the system quality of features such as performance, availability, scalability, and serviceability. QoS requirements are driven by business needs specified in the business requirements. **For example,** if services must be available 24 hours a day throughout the year, the availability requirement must address this business requirement.

The following table lists the system qualities that typically form a basis for QoS requirements:

System Quality	Description
Performance	The measurement of response time and throughput with respect to user load conditions.
Availability	A measure of how often a system's resources and services are accessible to end users, often expressed as the uptime of a system.
Scalability	The ability to add capacity (and users) to a deployed system over time. Scalability typically involves adding resources to the system but should not require changes to the deployment architecture.
Security	A complex combination of factors that describe the integrity of a system and its users. Security includes authentication and authorisation of users, security of data, and secure access to a deployed system.

Ques 26) Discuss the common techniques used in computer networks to improve the QoS. **(2018[04])**

Or

Explain any two methods to ensure QoS. **(2019[04])**

Ans: Common Techniques/Methods to Improve QoS

There exist techniques that can be used to improve the quality of service. The four common methods are:

1) **Scheduling:** Packets from different flows arrive at a switch or router for processing. A good scheduling technique treats the different flows in a fair and appropriate manner. Several scheduling techniques are designed to improve the quality of service. Some of them are:

i) **FIFO Queuing:** In first-in, first-out (FIFO) queuing, packets wait in a buffer (queue) until the node (router or switch) is ready to process them. If the average arrival rate is higher than the average processing rate, the queue will fill up and new packets will be discarded. A FIFO queue is familiar to those who have had to wait for a bus at a bus stop.

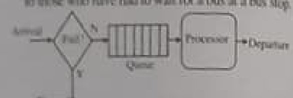


Figure 3.16: FIFO queue

ii) **Priority Queuing:** In priority queuing, packets are first assigned to a priority class. Each priority class has its own queue. The packets in the highest-priority queue are processed first. Packets in the lowest-priority queue are processed last. Note that the system does not stop serving a queue until it is empty. **Figure 3.17** shows priority queuing with two priority levels (for simplicity).

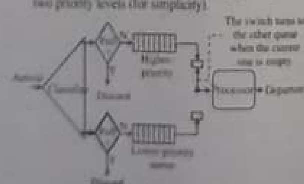


Figure 3.17: Priority queuing

A priority queue can provide better QoS than the FIFO queue because higher priority traffic, such as multimedia, can reach the destination with less delay. However, there is a potential drawback.

If there is a continuous flow in a high-priority queue, the packets in the lower-priority queues will never have a chance to be processed. This is a condition called starvation.

iii) **Weighted Fair Queuing:** A better scheduling method is weighted fair queuing. In this technique, the packets are still assigned to different classes and admitted to different queues.

The queues, however, are weighted based on the priority of the queues; higher priority means a higher weight. The system processes packets in each queue in a round-robin fashion with the number of packets selected from each queue based on the corresponding weight.

For example, if the weights are 3, 2, and 1, three packets are processed from the first queue, two from the second queue, and one from the third queue. If the system does not impose priority on the classes, all weights can be equal. In this way, we have fair queuing with priority. **Figure 3.18** shows the technique with three classes.

2) **Traffic Shaping:** Traffic shaping is a mechanism to control the amount and the rate of the traffic sent to the network. Two techniques can shape traffic:

i) Leaky bucket and
ii) Token bucket

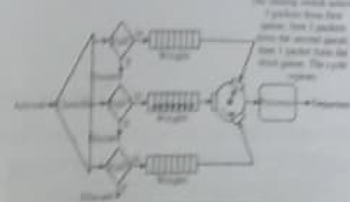


Figure 3.18: Weighted Fair Queuing

The two techniques can be combined to credit an idle host and at the same time regulate the traffic. The leaky bucket is applied after the token bucket; the rate of the leaky bucket needs to be higher than the rate of tokens dropped in the bucket.

3) **Resource Reservation:** A flow of data needs resources such as a buffer, bandwidth, CPU time, and so on. The quality of service is improved if these resources are reserved beforehand. We discuss in this section one QoS model called Integrated Services, which depends heavily on resource reservation to improve the quality of service.

4) **Admission Control:** Admission control refers to the mechanism used by a router, or a switch, to accept or reject a flow based on predefined parameters called flow specifications. Before a router accepts a flow for processing, it checks the flow specifications to see if its capacity (in terms of bandwidth, buffer size, CPU speed, etc.) and its previous commitments to other flows can handle the new flow.

Module 4

Network Layer in the Internet

NETWORK LAYER IN INTERNET

Ques 1) Give the introduction of TCP/IP protocol?

Or

What is internet protocol (IP)? Also give the frame format of IP?

Ans: TCP/IP Protocol

The TCP/IP holds the Internet together is the network layer protocol. The **Internet Protocol (IP)** provides all of the Internet's data transport services. Every other Internet protocol is ultimately either layered atop Internet Protocol, or used to support Internet Protocol from below.

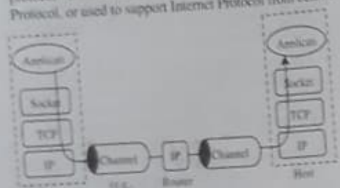


Figure 4.1: TCP/IP

The TCP/IP protocol suite, also known as the internet Protocols, is a suite of industry-standard protocols and can handle just about any task for the user.

Internet Protocol (IP)

Internet Protocol (IP) is a datagram-oriented protocol, treating each packet independently. Also Internet Protocol makes no attempt to determine if packets reach their destination or to take corrective action if they do not. Internet Protocol provides the following functions:

1) Addressing 2) Fragmentation 3) Packet timeouts

Internet Protocol (IP) is a network-layer (Layer 3) protocol that contains addressing information and some control information that enables packets to be routed. Along with the Transmission Control Protocol (TCP), IP represents the heart of the Internet protocols.

IP has two primary responsibilities:

1) Providing connectionless, best-effort delivery of datagrams through an internetwork; and
2) Providing fragmentation and reassembly of datagrams to support data links with different maximum-transmission unit (MTU) sizes.

IP Packet Format

An IP packet contains several types of information, as illustrated in **Figure 4.2**.

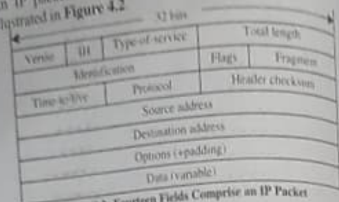


Figure 4.2: Fourteen Fields Comprise an IP Packet

1) **Version:** Indicates the version of IP currently used.
2) **IP Header Length (IHL):** Indicates the datagram header length in 32-bit words.
3) **Type-of-Service:** Specifies how an upper-layer protocol would like a current datagram to be handled, and assigns datagram various levels of importance.
4) **Total Length:** Specifies the length, in bytes, of the entire IP packet, including the data and header.
5) **Identification:** Contains an integer that identifies the current datagram. This field is used to help piece together datagram fragments.
6) **Flags:** Consists of a 3-bit field of which the two low-order (least-significant) bits control fragmentation. The low-order bit specifies whether the packet can be fragmented. The middle bit specifies whether the packet is the last fragment in a series of fragmented packets. The third or high-order bit is not used.
7) **Fragment Offset:** Indicates the position of the fragment's data relative to the beginning of the data in the original datagram, which allows the destination IP process to properly reconstruct the original datagram.
8) **Time-to-Live:** Maintains a counter that gradually decrements down to zero, at which point the datagram is discarded. This keeps packets from looping endlessly.
9) **Protocol:** Indicates which upper-layer protocol receives incoming packets after IP processing is complete.
10) **Header Checksum:** Helps ensure IP header integrity.
11) **Source Address:** Specifies the sending node.
12) **Destination Address:** Specifies the receiving node.
13) **Options:** Allows IP to support various options, such as security.
14) **Data:** Contains upper-layer information.

Ques 2) What is IP Addresses? Discuss its type.

Or

What is classful and classless addressing?

Or

Explain the IP frame format and IP address classes in detail.

Or

List the private IP address ranges of class A, B and C? (2019[03])

Ans: IP Addressing

As with any other network-layer protocol, the IP addressing scheme is integral to the process of routing IP datagrams through an internetwork. Each IP address has specific components and follows a basic format. These IP addresses can be subdivided and used to create addresses for sub networks.

Each host on a TCP/IP network is assigned a unique 32-bit logical address that is divided into two main parts:

1) **Network Number:** The network number identifies a network and must be assigned by the Internet Network Information Center (InterNIC) if the network is to be part of the Internet.
2) **Host Number:** The host number identifies a host on a network and is assigned by the local network administrator.

Types of IP Addressing

IP addressing can be two types:

1) **Classful Addressing:** In the classful addressing system all the IP addresses that are available are divided into the five classes A,B,C,D and E, in which class A, B and C address are frequently used because class D is for Multicast and is rarely used and class E is reserved and is not currently used. Each of the IP address belongs to a particular class that's why they are classful addresses.

Earlier this addressing system did not have any name, but when classless addressing system came into existence then it is named as Classful addressing system. The main disadvantage of classful addressing is that it limited the flexibility and number of addresses that can be assigned to any device.

One of the major **disadvantages of classful addressing** is that it does not send subnet information but it will send the complete network address. The router will supply its own subnet mask based on its locally configured subnets.

IP Address Format

The 32 bits IP address is divided into four octets and each octet is written in eight bit decimal numbers. These four octets are separated by dots and ranges from 0 to 255. The binary weights of each bit in the octet are 128, 64, 32, 16, 8, 4, 2, 1. The format of the 32-bit IP address is illustrated in the **figure 4.3**.

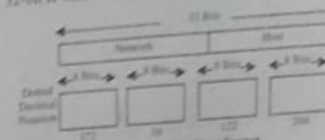


Figure 4.3: IP Address Format

IP Address Classes

A class is used to recognise the part of 'network address' and 'node address' given in an IP address.

There are five classes associated with IP addresses: A, B, C, D and E where only A, B and C are used for commercial purpose.

The network class can be determined by examining the left most bits of the network address.

The First octet from left of IP address constitute the network address of class A address, where First two octets from the network address of class B address and First 3 octets from the left constitute the network address of class C address.

The reference information of the five address classes are defined in the following **table 4.1**.

Table 4.1: Reference Information about Five IP Address Classes

IP Address Class	Format	Objective	High order Bit(s)	Address Range	No. of Bits Network/Host	Maximum Hosts
A	N.H.H.H	Few large organization	0	1.0.0.0. to 127.0.0.0	7/24	16,777, 216 (2^{24}-2)
B	N.N.H.H	Medium-size organization	1,0	128.1.0.0 to 191.254.0.0	14/16	65,536 (2^{16}-2)
C	N.N.N.H	Relatively small organization	1, 1, 0	192.0.1.0 to 223.255.254.0	22/8	256 (2^{8}-2)
D	N/A	Multicast groups (RFC 1112)	1, 1, 1, 0	224.0.0.0 to 239.255.255.255	N/A (not for commercial use)	N/A
E	N/A	Experimental	1, 1, 1, 1	240.0.0.0 to 240.255.255.255	N/A	N/A

Where, N = Network number, H = Host number. For recognising the class of IP address examine the first octet of address and match it with already fixed range of the class. The following **table** illustrates the range of the various classes:

Address Class	First Octet in Decimal	High-Order Bits
Class A	1 – 126	0
Class B	128 – 191	10
Class C	192 – 223	110

For example, consider an IP address 172.31.1.2. Its first octet is 172, so, it belongs to class B.

2) **Classless Addressing:** There were certain problems with classful addressing such as address depletion and less organisation access to Internet. To overcome these problems, classful addressing is replaced with classless addressing. As the name of the addressing scheme implies, the addresses are not divided into classes; however, they are divided into blocks and the size of blocks varies according to the size of entity to which the addresses are to be allocated. For example, only a few addresses may be allocated to a very small organisation while a larger organisation may obtain thousands of addresses IPv6 addressing is a classless addressing.

The Internet authorities have enforced certain limitations on classless address blocks to make the handling of addresses easier. These **limitations** are as follows:

i) The addresses of a block must be contiguous.
ii) Each block must have a power of 2(that is, 1, 2, 4, 8....) number of addresses.
iii) The first address in a block must be evenly divisible by the total number of addresses in that block.

Ques 3) Compare classful and classless addressing, giving examples for both. (2018[03])

Ans: Comparison between Classful and Classless Addressing

Classful Addressing	Classless Addressing
Addresses have 3 parts: network, subnet and host	Addresses have 2 parts subnet or prefix and host
Does not advertise masks not support VLSM, RIP-1 and IGRP	Does advertise masks and supports VLSM, RIP2, EIGRP and OSPF
IP forwarding process is restricted in how it uses the default route	IP forwarding process has not restrictions on how it uses the default route
In classful addressing, the network information portion of an IPv4 address (the network ID) is limited to the first 8 bits in a Class A address, the first 16 bits in a Class B address, and the first 24 bits in class C address. Host information is contained in the last 24 bits for a Class A address, the last 16 bits in a Class B address, and the last 8 bits in a Class C address. **For example,** some IPv4 addresses separated into network and host information according to the classful addressing convention: 1) Class A network Address 114.56.204.33 Network Information = 114 Host Information = 56.204.33 2) Class B network address 147.12.38.81 Network Information = 147.12 Host Information = 38.81 3) Class C Network Address 214.57.42.7 Network Information = 214.57.42 Host Information = 7	**For example,** let assume an organization was given a class A block as 73.0.0.0 in the past. If the block is not revoked by the authority, the classless architecture assumes that the organization has a block 73.0.0.0/8 in classless addressing

Ques 4) What is Subnetting?

Or

What is subnet mask?

Or

Define Subnetting. What are the advantages of Subnetting? Explain with an example. (2018[03])

Or

Illustrate subnetting with an example. (2020[04])

Ans: Subnetting

Subnetting is a unique and powerful feature that is exclusive to the TCP/IP protocol and is one of the reasons TCP/IP offers great scalability. Subnetting allows network address to be further divided, apart from the already established classful boundaries, into smaller, more manageable networks. This division provides for unparalleled scalability and hierarchy, and gives a network administrator benefits such as reduced network traffic, less susceptibility to broadcast traffic, network optimisation, and greater ease of management. **For example,** if you were to borrow one bit from the host portion of a Class B network, your subnet mask would be 255.255.128.0.

Remember, you borrow bits from left to right, but those bit positions still hold their original values, so the rightmost bit would be valued at 128. If you wanted to create four subnets, your subnet mask would read 255.255.192 because you have now borrowed two bits, one valued at 128 and another valued at 64. So, 128 + 64 = 192.

Subnet Mask

There are two parts to the IP address, the network portion and the host portion. Node assigned that IP address as well as other nodes that must communicate with it have no idea of the location of the line between host and network portions of the address. The subnet mask provides the answer to this dilemma. The subnet mask follows the IP address and details the line indicating where the network portion of the address ends and the host portion begin. Like the IP address, the subnet mask is in a 4-octet, 32-byte format. An **example** of a subnet mask is 255.0.0.0, a value of 255 means match all. Each of the three configurable IP address classes has a default subnet mask:

1) Class A 255.0.0.0
2) Class B 255.255.0.0
3) Class C 255.255.255.0

Advantages of Subnetting

1) Minimizes the network traffic through decreasing the volume of broadcasts.
2) Increases addressing flexibility.

3) Increases the number of allowed hosts in local area network.
4) The network security can be readily employed between subnets rather than employing it in the whole network.
5) Subnets are easy to maintain and manage.

Ques 5) Find the class of each address:

1) **4.23.145.90** 2) **227.34.78.7** 3) **246.7.3.8**
4) **29.6.8.4** 5) **198.76.9.23**

Ans: The first byte defines the class.

1) Class A 2) Class D 3) Class E
4) Class B 5) Class C

Ques 6) IP address 172. 31. 192. 166 and subnet mask 255. 255. 255. 248, which subnet does the IP address belong.

Ans:

Host id is .248
i.e. it uses 5 bit
i.e. host = 2^5 = 32, i.e.

I[st]	subnet network address	172. 31. 192. 0
II[nd]	subnet network address	172. 31. 192. 32
III[rd]	subnet network address	172. 31. 192. 64
IV[th]	subnet network address	172. 31. 192. 96
V[th]	subnet network address	172. 31. 192. 128
VI[th]	subnet network address	172. 31. 192. 160
VII[th]	subnet network address	172. 31. 192. 192
VIII[th]	subnet network address	172. 31. 192. 224

So the IP address 172.31.192.166 lies in the VI[th] subnet network then the subnet network is 172.31.192.160

Ques 7) Subnet the Class C IP address 206.16.2.0 so that you have 30 subnets. What is the subnet mask for the maximum number of hosts? How many hosts can each subnet have?
(2019[03])

Ans: Current mask= 255.255.255.0
Bits needs for 30 subnets =5 =2^5 = 32 possible subnets
Bits left for hosts = 3 = 2^3 = 8 − 2 = 6 possible hosts.
So mask in binary = 11111000= 248 decimal
Final Mask = 255.255.255.248
Address of host 3 on subnet 2 is
Subnet 2 = 00010000 host 3 = 00000011
Add the two together =00010011=19
Therefore IP address of host 3 on subnet 2
= 206.11.2.19

Ques 8) How do you Subnet the Class C IP Address 195.1.1.0 So that you have 10 subnets each with a maximum 12 hosts on each subnet.

Ans: Current mask= 255.255.255.0
Bits needs for 10 subnets =4 =24 =16 possible subnets
Bits needs for 12 hosts = 4 = 24 = 16-2=14 possible hosts

So our mask in binary =11110000= 240 decimal
Final Mask =255.255.255.240

Ques 9) A network on the Internet has a subnet mask of 255.255.240.0. What is the maximum number of hosts it can handle?

Ans: Subnet Mask : - 255.255.240.0

$$\underbrace{11111111\;.\;11111111\;.\;1111}_{\text{net id}}\underbrace{0000\;.\;00000000}_{\text{host id}}$$

It is a class B network. For a class B network, the upper 16 bits form the network address and lower 16 bits are subnet and host fields. In lower 16 bits most significant 4 bits are 1111.This leaves 12 bits for the host number.So,4096(212) host address exists. First and Last address are special so the maximum number of address = 4096 − 2 = 4094

INTERNET CONTROL PROTOCOLS

Ques 10) What are the internet control protocols? List them.

Ans: Internet Control Protocols

At the network layer (or, more accurately, the internetwork layer), TCP/IP supports the internetwork protocol (IP). IP contains four supporting protocols:

1) Address Resolution Protocol(ARP)
2) Reverse Address Resolution Protocol (RARP)
3) Internet Control Message Protocol(ICMP)
4) Internet Group Message Protocol(IGMP)
5) BOOTP

Ques 11) Discuss about Internet Control Message Protocol (ICMP)?

Or

Explain the role of ICMP. (2019[04])

Or

Explain ICMP. (2020[2.5] [3])

Or

Explain ICMP in detail with advantages and disadvantages. (2020[05])

Ans: Internet Control Message Protocol (ICMP)

The internet control message protocol (ICMP) is a mechanism used by hosts and routers to send notification of datagram problems back to the sender. ICMP uses echo test/reply to test whether a destination is reachable and responding. It also handles both control and error messages, but its sole function is to report problems, not correct them. Responsibility for correction lies with the sender.

A datagram carries only the addresses of the original sender and the final destination. It does not know the addresses of the previous router(s) that passed it along.

For this reason, ICMP can send messages only to the source, not to an intermediate router. ICMP is often considered part of the IP layer.

It communicates error messages and other conditions that require attention. ICMP messages are usually acted on by either the IP layer or the higher layer protocol (TCP or UDP). Some ICMP messages cause errors to be returned to user processes. ICMP messages are transmitted within IP datagrams, as shown in **figure 4.4**:

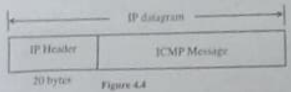


Figure 4.4

Figure 5.2 shows the format of an ICMP message. The first 4 bytes have the same format for all messages, but the remainder differs from one message to the next.

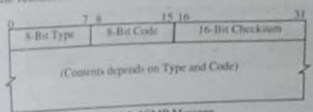


Figure 4.5: ICMP Message

There are 15 different values for the type field, which identify the particular ICMP message. Some types of ICMP messages then use different values of the code field to further specify the condition.

The checksum field covers the entire ICMP message. The ICMP checksum is required.

Advantages of ICMP

1) ICMP protocol helps network administrators by assisting them in diagnosing networking issues. Most issues that arise, like server outages or computer failure, are determined with two helpful commands. These commands are PING and TRACERT.
2) Network speed provides users with the access on demand that they require in order to accomplish their task on the network or Internet.
3) Every network has multiple layers that actually make up the entire network, from the computers and servers that operate on the network, to even the pieces you do not see--like the Network layer which helps ICMP protocol actually function. The network layer builds the backbone of the Internet and all networks that transfer any type of data requests.

Disadvantages of ICMP

1) If a packet does not match any route and there is no default route in the routing table, the device sends a Network Unreachable ICMP error packet to the source.
2) If a packet is destined for the device but the transport layer protocol of the packet is not supported by the device, the device sends a Protocol Unreachable ICMP error packet to the source.

3) If a UDP packet is destined for the device but the packet's port number does not match the corresponding process, the device sends the source a Port Unreachable ICMP error packet.

Ques 12) Discuss how error reporting happens in ICMP?

Or

List and explain the different types of error reporting messages used by ICMP. **(2018[03])**

Ans: Error Reporting

ICMP always reports error messages to the original source. Five types of errors are handled (**figure 4.6**):

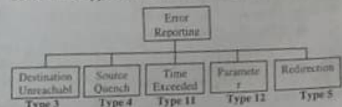


Figure 4.6: Error Reporting Messages

1) **Destination Unreachable:** The message of "Destination unreachable" is passed to the sender when the receiver could not be contacted, or the packet was discarded because the ultimate destination could not be contacted.
2) **Source Quench:** It is a message from one host to another asking the other host to slow down the speed at which the packets are being sent. Source Quench is one of the ways to control the packet flow on the internet.
3) **Time Exceeded:** Also known as TTL Time Exceeded, this is an interesting message generated using ICMP. On the basic level, all the packets transmitted through the internet world will have a TTL value. TTL basically stands for "Time to Live". It is a like parameter which decides how long a packet should live before it would be discarded.
4) **Parameter Problem:** Sometimes, problems might not specifically be covered by any ICMP messages. In that case, Parameter Problem is shown.
5) **ICMP Redirects:** ICMP redirect messages direct a host to deliver the next packet for the same destination IP address to a different router.

ICMP forms an error packet, which is then encapsulated in an IP datagram (**figure 4.7**).

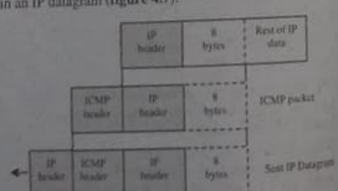


Figure 4.7: Contents of Data Field for the Error Messages

Ques 13) Explain about the ICMP timestamp request and reply?

Ans: ICMP Timestamp Request and Reply

The ICMP timestamp request allows a system to query another for the current time. The recommended value to be returned is the number of milliseconds since midnight coordinated Universal Time.

The **advantage** of this ICMP message is that it provides millisecond resolution, whereas some other methods for obtaining the time from another host (such as the rdate command provided by some Unix systems) provide a resolution of seconds.

The **drawback** is that only the time since midnight is returned – the caller must know the date from some other means.

Figure 4.8 shows the format of the ICMP timestamp request and reply messages. The requestor fills in the originate timestamp and sends the request. The replying system fills in the receive timestamp when it receives the request, and the transmit timestamp when it sends the reply.

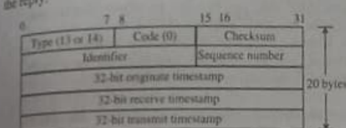


Figure 4.8: ICMP Timestamp Request and Reply Messages

In actuality, however, most implementations set the latter two fields to the same value. (The reason for providing the three fields is to let the sender compute the time for the request to be sent and separately compute the time for the reply to be sent).

Ques 14) What is Address Resolution Protocol (ARP)? Explain its working.

Or

What is the use of ARP? Explain ARP operation and packet format. **(2018[07])**

Or

Define address resolution problem. **(2019[03])**

Ans: Uses of Address Resolution Protocol (ARP)

ARP is used to find the physical address of the node when its Internet address is known. Anytime a host, or a router needs to find the physical address of another host on its network, it formats an ARP query packet that includes the IP address and broadcasts in over the network.

Every host on the network receives and processes the ARP packet, but only the intended recipient recognizes its internet address and sends back its physical address. The host holding the datagram adds the address of the target host both to its cache memory and to the datagram header, then sends the datagram on its way.

ARP is a low level protocol that uses the services of the MAC (Data Link) Layer, and as with all protocols, is then encapsulated in a physical network frame.

In this case the Source Address field of the physical frame will indicate the station that is requesting the address resolution, while the Destination Address field will contain the broadcast address.

Where a Type field is present, this will contain a code to indicate the ARP protocol, so that receiving stations will be able to correctly process the frame. **For example,** in the case of Ethernet, the Type field will contain 0x0806.

Working of Address Resolution Protocol (ARP)/ARP Operation

Step 1: When a source device want to communicate with another device, source device checks its Address Resolution Protocol (ARP) cache to find it already has a resolved MAC address of the destination device.

If it is there, it will use that address for communication. To view your Local Address Resolution Protocol (ARP) cache, Open Command Prompt and type command "arp a" (Without double quotes using Windows Operating Systems).

Step 2: If ARP resolution is not there in local cache, the source machine will generate an Address Resolution Protocol (ARP) request message, it puts its own data link layer address as the Sender Hardware Address and its own IP address as the Sender Protocol Address. It fills the destination IP address as the Target Protocol Address. The Target Hardware Address will be left blank, since the machine is trying to find that.

Step 3: The source broadcast the Address Resolution Protocol (ARP) request message to the local network.

Step 4: The message is received by each device on the LAN since it is a broadcast. Each device compare the Target Protocol Address (IP Address of the machine to which the source is trying to communicate) with its own Protocol Address (IP Address). Those who do not match will drop the packet without any action.

Step 5: When the targeted device checks the Target Protocol Address, it will find a match and will generate an Address Resolution Protocol (ARP) reply message. It takes the Sender Hardware Address and the Sender Protocol Address fields from the Address Resolution Protocol (ARP) request message and uses these values for the Targeted Hardware Address and Targeted Protocol Address of the reply message.

Step 6: The destination device will update its Address Resolution Protocol (ARP) cache, since it need to contact the sender machine soon.

Step 7: Destination device send the Address Resolution Protocol (ARP) reply message and it will not be a broadcast, but a unicast.

Step 8: The source machine will process the Address Resolution Protocol (ARP) reply from destination, it store the Sender Hardware Address as the layer 2 address of the destination.

Step 9: The source machine will update its Address Resolution Protocol (ARP) cache with the Sender Hardware Address and Sender Protocol Address it received from the Address Resolution Protocol (ARP) reply message.

ARP Packet Format

The format of ARP packet is shown in **Figure 4.9**. The ARP packet comprise various field, which are described as follows.

1) **Hardware Type:** It is a 16-bit long field that defines the type of the network on which ARP is running. For example, if ARP is running on Ethernet then the value of this field will be one ARP can be used on physical network.

2) **Protocol Type:** It is a 16-bit long field that defines the protocol used by ARP. For example, if ARP is using IPv4 protocol then the value of this field will be $(0800)_{16}$. ARP can be used with any protocol.

3) **Hardware Length:** It is an 8-bit long field that defines the length of MAC address in byte.

Hardware type 16 bits		Protocol type 16 bits
Hardware length 8 bits	Protocol length 8 bits	Operation 16 bits
Sender hardware address		
Sender protocol address		
Target hardware address		
Target protocol address		

Figure 4.9 ARP Packet Format

4) **Protocol Length:** It is an 8-bit long field that defines the length of address in bytes.

5) **Operation:** It is a 16-bit long field that defines the type of packet being carried out. For ARP request packet the value of this field will be one and for ARP response packet, the value will be two.

6) **Sender Hardware Address:** It is a variable-length field that defines the MAC address of the sender node.

7) **Sender Protocol Address:** It is a of variable-length field that defines the IP address of the sender node.

8) **Target Hardware Address:** It is a variable-length field that defines the MAC address of the destination node. In case of an ARP request packet, the value of this field is 0, as the MAC address of the receiver node is not known to the sender node.

9) **Target Protocol Address:** It is a variable-length field that defines the IP address of the destination node.

Ques 15) What do you understand by Gratuitous and proxy ARP?

Ans: Gratuitous ARP

Gratuitous ARP is used when a node (end system) has selected an IP address and then wishes to defend its chosen address on the local area network (i.e. to check no other node is using the same IP address).

It can also be used to force a common view of the node's IP address (e.g. after the IP address has changed). Use of this is common when an interface is first configured, as the node attempts to clear out any stale caches that might be present on other hosts. The node simply sends an ARP request for itself.

Proxy ARP

Proxy ARP is the name given when a node responds to an ARP request on behalf of another node. This is commonly used to redirect traffic sent to one IP address to another system.

Proxy ARP can also be used to subvert traffic away from the intended recipient. By responding instead of the intended recipient, a node can pretend to be a different node in a network, and therefore force traffic directed to the node to be redirected to itself.

Ques 16) Define Reverse Address Resolution Protocol (RARP)?

Or

Explain RARP. (2019[03]) (2020[2.5])

Ans: Reverse Address Resolution Protocol (RARP)

RARP works much like ARP. The host wishing to retrieve its internet address broadcasts an RARP query packet that contains its physical address to every host on its physical network. A server on the network recognizes the RARP packet and returns the host's internet address.

The TCP/IP protocol that allows a computer to obtain its IP address from a server is known as the Reverse Address Resolution Protocol (RARP). RARP is adapted from the ARP protocol and uses the same message format. Like an ARP message, a RARP message is sent from one machine to another encapsulated in the data portion of a network frame.

For example, an Ethernet frame carrying a RARP request has the usual preamble, Ethernet source and destination addresses, and packet type fields in front of the frame. The frame type contains the value 8035 to identify the contents of the frame as a RARP message. The data portion of the frame contains the 28-octet RARP message.

Figure 5.7 shows how a host uses RARP. The sender broadcasts a RARP request that specifies itself as both

the sender and target machine, and supplies its physical network address in the target hardware address field. All computers on the network receive the request, but only those authorized to supply the RARP service process the request and send a reply; such computers are known informally as RARP servers. For RARP to succeed, the network must contain at least one RARP server:

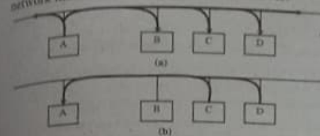


Figure 4.10: Example Exchange using the RARP Protocol, (a) Machine A Broadcasts a RARP Request specifying itself as a Target, and (b) Those Machines Authorized to Supply the RARP Service (C and D) Reply Directly to A.

Servers answer requests by filling in the target protocol address field, changing the message type from request to reply, and sending the reply back directly to the machine making the request. The original machine receives replies from all RARP servers, even though only the first is needed.

Ques 17) Write short note on BOOTP.

Or

Give the importance of BOOTP. (2019[04])

Ans: BOOT Strap Protocol (BOOTP)

To overcome some of the drawbacks of RARP, researchers developed the BOOT strap Protocol (BOOTP). Later, the Dynamic Host Configuration Protocol (DHCP) was developed as a successor to BOOTP because the two protocols are closely related.

Because it uses UDP and IP, BOOTP can be implemented with an application program. Like RARP, BOOTP operates in the client-server paradigm and requires only a single packet exchange. However, BOOTP is more efficient than RARP because a single BOOTP message specifies many items needed at start-up, including a computer's IP address, the address of a router, and the address of a server.

BOOTP also includes a vendor-specific field in the reply that allows hardware vendors to send additional information used only for their computers. BOOTP places all responsibility for reliable communication on the client. Because UDP uses IP for delivery, messages can be delayed, lost, delivered out of order, or duplicated. Furthermore, because IP does not provide a checksum for data, the UDP datagram could arrive with some bits corrupted. To guard against corruption, BOOTP requires that UDP use checksums. It also specifies that requests and replies should be sent with the do not fragment bit set to accommodate clients that have too little memory to re-assemble datagrams. BOOTP is also constructed to allow multiple replies; it accepts and processes the first.

To handle datagram loss, BOOTP uses the conventional technique of timeout and re-transmission. When the client transmits a request, it starts a timer.

If no reply arrives before the timer expires, the client must re-transmit the request. Of course, after a power failure all machines on a network will re-boot simultaneously, possibly over-running the BOOTP server(s) with requests.

If all clients use exactly the same re-transmission timeout, many or all of them will attempt to re-transmit simultaneously. To avoid the resulting collisions, the BOOTP specification recommends using a random delay. In addition, the specification recommends starting with a random timeout value between 0 and 4 seconds, and doubling the timer after each re-transmission.

After the timer reaches a large value, 60 seconds, the client does not increase the timer, but continues to use randomization. Doubling the timeout after each re-transmission keeps BOOTP from adding excessive traffic to a congested network; the randomization helps avoid simultaneous transmissions.

Ques 18) What is DHCP? Discuss the DHCP header with diagram.

Ans: Dynamic Host Configuration Protocol (DHCP)

DHCP is a protocol used to assign an IP address to a computer or device connected to a network automatically. Routers, switches, or servers that assign addresses to other computers using DHCP on a network make setup and management of the network easier by not requiring the network admin to define each address for each computer and network device on the network.

Dynamic Host Configuration Protocol (DHCP) provides dynamic configuration information to hosts running the Internet protocol. DHCP is based on a client/server model whereby a client requests and receives configuration information from a server, which allows it to operate properly over the IP network.

DHCP is useful for automatic configuration of client network interfaces. When configuring the client system, the administrator chooses DHCP instead of specifying an IP address, gateway, or DNS servers. The client retrieves this information from the DHCP server.

DHCP is also useful if an administrator wants to change the IP addresses of a large number of systems. Instead of reconfiguring all the systems, one can just edit one DHCP configuration file on the server for the new set of IP addresses.

If the DNS servers for an organization changes, the changes are made on the DHCP server, not on the DHCP clients. When the administrator restarts the network or reboots the clients, the changes will go into effect.

DHCP Header

Figure 4.11 shows the DCHP header:

32 bits

8	8	8	8
Operation	Hardware Type	Hardware Length	Hops
Transaction ID (xid)			
Seconds		Flags	
Client Internet Address (ciaddr)			
Internet Address (yiaddr)			
Server Internet Address (siaddr)			
Gateway Internet Address (giaddr)			
Client Hardware Address (chaddr)			
Server Host Name			
Boot File (optional)			
Options (optional)			

Figure 4.11: DHCP Header

The fields in the header are as follows:

1) **Operation:** Message operation code (1 = BootRequest, 2 = BootReply).
2) **Hardware Type:** Hardware address type (Ethernet, Token-Ring, and so on).
3) **Hardware Length:** Hardware address length (e.g., 6 bytes for Ethernet).
4) **Hops:** Client sets this to zero. This is used optionally by relay agents.
5) **Transaction ID (XID):** Random number chosen by the client to associate messages and responses sent between a client and server.
6) **Seconds:** Seconds elapsed since client began address acquisition or renewal process.
7) **Flags:** Set to indicate if a client can receive unicast frames (0 = can accept unicast, 1 = can accept only broadcasts).
8) **Client Internet Address:** Filled in if client is in the BOUND, RENEW, or REBINDING state.
9) **Client Internet Address:** IP address of client.
10) **Server Internet Address:** IP address of DHCP server.
11) **Gateway Internet Address:** Relay agent IP address (usually a router).
12) **Client Hardware Address:** Hardware (MAC) address of client.
13) **Server Host Name:** Optional host name of DHCP server.
14) **Boot File:** Optional name of boot file (if requested by client).
15) **Options:** Optional parameters field.

Ques 19) Write short note on the following:
- **DHCP Messages**
- **DHCP Process**

Ans: DHCP Messages

RFC 2131 specifies the following DHCP messages:

1) **DHCPDISCOVER:** Client broadcast to locate available servers.
2) **DHCPOFFER:** Server to client in response to DHCPDISCOVER with offer of configuration parameters.
3) **DHCPREQUEST:** Client message to servers, doing one of the following:
   i) Requesting offered parameters from one server and implicitly declining offers from all others;
   ii) Confirming correctness of previously allocated address after, e.g., a system reboot; and
   iii) Extending the lease on a particular network address.
4) **DHCPACK:** Server to client with configuration parameters, including a committed network address.
5) **DHCPNAK:** Server to client indicating client's notion of network address is incorrect (e.g., client has moved to new subnet) or client's lease has expired.
6) **DHCPDECLINE:** Client to server, indicating that a network address is already in use.
7) **DHCPRELEASE:** Client to server, relinquishing network address and canceling the remaining lease.
8) **DHCPINFORM:** Client to server, asking only for local configuration parameters; client already has externally configured network address. The first four messages make-up the standard DHCP four-packet process.

Ques 20) Differentiate between BOOTP and DHCP. (2018[05])

Ans: Difference between BOOTP and DHCP

Basis	BOOTP	DHCP
Autoconfiguration	Not possible only supports manual configuration.	It automatically obtains and assigns IP addresses.
Temporary IP addressing	Not provided	Provided for a limited amount of time.
Compatibility	Not compatible with DHCP clients.	Interoperable with the BOOTP clients.
Mobile machines	IP Configuration and information access are not possible.	Supports mobility of machines.
Error occurrence	Manual configuration is prone to errors.	Autoconfiguration is immune to errors.
Usage	Provides the information to the diskless computer or workstation.	It requires disks to store and forward the information.

Ques 21) What is Internet multicasting? What are the applications of multicasting?

Ans: Multicasting

In multicast communication, there is one source and a group of destinations. The relationship is one-to-many. In this type of communication, the source address is a unicast address, but the destination address is a group address, which defines one or more destinations.

In multicasting, the router may forward the received packet through several of its interfaces.

Applications of Multicasting

1) Access to Distributed Databases
2) Information Dissemination
3) Dissemination of News
4) Teleconferencing
5) Distance Learning

Ques 22) What are the different types of routing protocols? Explain.

Or

What is interior and exterior routing protocol?

Ans: Types of Routing Protocol

Internet routing can be defined more precisely. All Internets routing protocols fall into one of the two categories:

1) **Interior Gateway Protocols (IGPs):** The router within an autonomous system uses an Interior Gateway Protocol (IGP) to exchange routing information. There are several IGPs available; each autonomous system is free to choose its own IGP. Usually, an IGP is easy to install and operate, but an IGP may limit the size or routing complexity of an autonomous system. There are two types Interior Gateway Protocols:
   i) **Interior gateway protocols type 1**, link-state routing protocols, such as Open Shortest Path First (OSPF) and IS-IS.
   ii) **Interior gateway protocols type 2**, distance-vector routing protocols, such as Routing Information Protocol, RIPv2, IGRP. **Enhanced Interior Gateway Routing Protocol (EIGRP)** is an advanced distance-vector routing protocol that is used on a computer network for automating routing decisions and configuration.
2) **Exterior Gateway Protocols (EGPs):** A router is one autonomous system uses an exterior gateway Protocol (EGP) to exchange routing information with a router in another autonomous system. EGPs are usually more complex to install and operate than IGPs, but EGPs offer more flexibility and lower overhead (i.e., less traffic). To save traffic, an EGP summarises routing information from the autonomous system before passing it to another autonomous system. More important, an EGP implements policy constraint that allows a system manager to determine exactly what information is released outside the organisation.

Exterior gateway protocols are routing protocols used on the Internet for exchanging routing information between Autonomous Systems, such as Border Gateway Protocol (BGP), Path Vector Routing Protocol.

Ques 23) Discuss about the Open Shortest Path First (OSPF) with suitable diagram.

Or

Discuss Open Shortest Path First (OSPF) with an example. (2020[04])

Ans: Open Shortest Path First (OSPF)

OSPF stands for Open Shortest Path First which uses link-state routing algorithm. Using the link state information which is available in routers, it constructs the topology in which the topology determines the routing table for routing decisions. It supports both variable-length subnet masking and classless inter-domain routing addressing models.

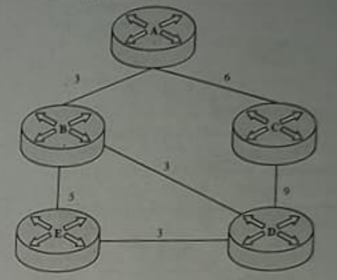


Figure 4.12: Simple Structure of OSPF

Since, it uses Dijkstra's algorithm, it computes the shortest path tree for each route. The main advantages of the OSPF (Open Shortest Path first) is that it handles the error detection by itself and it uses multicast addressing for routing in a broadcast domain.

SPF Calculation

Before running the calculation, it is required that all routers in the network to know about all the other routers in the same network and the links among them. The next step is to calculate the shortest path between each single router. For all the routers they exchange link-states which would be stored in the link-state database. Every time a router receives a link-state update, the information stores into the database and this router propagate the updated information to all the other routers. Below, is a simple model of how the SPF algorithm works.

A simple network formed by five routers; all the routers know about all the other routers and links. After all the paths are figured out, the path information are stored in the link database. The link database for the above model is : [A, B, 3], [A, C, 6], [B, A, 3], [B, D, 3], [B, E, 5], [C, A, 6], [C, D, 9], [D, C, 9], [D, B, 3], [D, E, 3] , [E, B, 5] and [E, D, 3].

Each term is referred to the originating router, the router connected to and the cost of the link between the two routers. Once the database of each router is finished, the router determines the Shortest Path Tree to all the destinations. (The shortest path in the SPF algorithm is called the Shortest Path Tree). The Dijkstra Shortest Path First is then running to determine the shortest path from a specific router to all the other routers in the network. Each router is put at the root of the Shortest Path Tree and then the shortest path to each destination is calculated. The accumulated cost to reach the destination would be the shortest path.

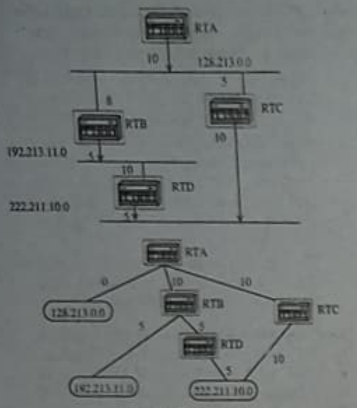


Figure 4.13: Shortest Path Tree

The cost (metric) of OSPF is the cost of sending packets across a certain interface. The formula to calcite the cost is:

cost = 10000 0000 /bandwidth in bps.

If the bandwidth is wider, the cost would be lower.

Above is a figure 4.13 of the structure used to calculate the Shortest Path Tree? When the Shortest Path Tree is completed, the router will work on the routing table.

Ques 24) What is BGP? What are the main characteristics of BGP? Also discuss its type.

Ans: Border Gateway Protocol (BGP)
BGP is a complex, advanced distance Exterior Gateway Protocol (EGP). BGP exchange routing information between Autonomous Systems (ASs).

BGP is especially used for exchanging routing information between all of the major Internet Service Providers (ISPs), as well between larger client sites and their respective ISPs. And, in some large enterprise networks, BGP is used to interconnect different geographical or administrative regions.

The Border Gateway Protocol (BGP) was developed for use in conjunction with internets that employ the TCP/IP suite, although the concepts are applicable to any internet.

BGP has become the preferred exterior router protocol for the internet. Functions BGP was designed to allow routers, called gateways in the standard, in different Autonomous Systems (ASs) to cooperate in the exchange of routing information.

The protocol operates in terms of messages, which are sent over TCP connections. BGP is primarily used to support the complexity of the public Internet; Cisco has added several clever and useful features to its BGP implementation (BGP 4).

Characteristics of BGP
1) It is an advanced distance-vector protocol.
2) BGP sends full routing updates at the start of the session, trigger updates are sent afterward.
3) BGP maintains connection by sending periodic keepalives.
4) It creates and maintains connections between peers, using TCP port 179.
5) BGP sends a triggered update when a keepalive, an update, or a notification is not received.
6) It has its own routing table, although it is capable of both sharing and inquiring of the interior IP routing table.
7) BGP uses a very complex metric, and is the source of its strength. The metric, referred to as attributes, allows great flexibility in path selection.

Types of BGP
There are two types of BGP:
1) **iBGP:** Internal BGP (iBGP) operates inside an autonomous System (AS).
2) **eBGP:** External BGP (eBGP) is also known as an interdomain routing protocol, operates outside an AS and connects one AS to another. These terms are just used to describe the same protocol just the area of operation is what differs.

Ques 25) Discuss about the BGP message format.
Or
Explain the following:
1) **Open Messages and Update Messages**
2) **Keepalive Messages and Notification Messages**

Ans: BGP Messages
Figure 4.14 illustrates the formats of all of the BGP messages.

Each message begins with a 19-octet **header containing** three fields (shaded area):
1) **Marker:** Reserved for authentication. The sender may insert a value in this field that would be used as part of an authentication mechanism to enable the recipient to verify the identity of the sender.
2) **Length:** Length of message in octets.
3) **Type:** Type of message – Open, Update, Notification, Keepalive.

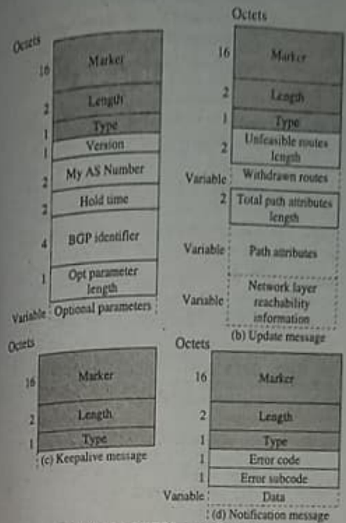


Figure 4.14: BGP Message Formats

The four types of messages are as below:
1) **Open Messages:** After a TCP connection is established between two BGP systems, they exchange BGP open messages to create a BGP connection between them. Once the connection is established, the two systems can exchange BGP messages and data traffic. Open messages consist of the BGP header plus the following fields:
   i) **Version:** The current BGP version number is 4.
   ii) **My AS Number:** BGP open message's AS number field contains 16-bits that contains the AS number of the BGP routing instance that transmitted the open message.
   iii) **Hold Time:** Proposed hold-time value.
   iv) **BGP Identifier:** IP address of the BGP system. This indicates the ID of the sender of the BGP open message and is equal to the IP address that is assigned to the device.
   v) **Optional Parameters Length:** The BGP open message's optional parameters length is an 8-bit field that indicates the number of bytes in the optional parameters section of the BGP open message.
   vi) **Optional Parameters:** The BGP open message's optional parameters contain all optional parameters for BGP sessions.
2) **Update Messages:** BGP systems send update messages to exchange network reachability information. BGP systems use this information to construct a graph that describes the relationships among all known ASs.

Update messages consist of the BGP header plus the following optional fields:
   i) **Unfeasible Routes Length:** Length of the withdrawn routes field.
   ii) **Withdrawn Routes:** IP address prefixes for the routes being withdrawn from service because they are no longer deemed reachable.
   iii) **Total Path Attribute Length:** Length of the path attributes field; it lists the path attributes for a feasible route to a destination.
   iv) **Path Attributes:** Properties of the routes, including the path origin, the multiple exit discriminator (MED), the originating system's preference for the route, and information about aggregation, communities, confederations, and route reflection.
   v) **Network Layer Reachability Information (NLRI):** IP address prefixes of feasible routes being advertised in the update message.
3) **Keepalive Messages:** This is the packet used to keep the session running when there are no updates. Keepalives are sent between BGP speakers to let each other know they are still there. When a BGP router fails to hear a Keepalive message, it removes all routes heard from that peer from its forwarding information base (FIB).
4) **Notification Messages:** The Notification Message is sent when an error condition is detected. The following errors may be reported:
   i) **Message Header Error:** It includes authentication and syntax errors.
   ii) **Open Message Error:** It includes syntax errors and options not recognised in an Open message. This message can also be used to indicate that a proposed Hold Time in an Open message is unacceptable.
   iii) **Update Message Error:** It includes syntax and validity errors in an Update message.
   iv) **Hold Timer Expired:** If the sending router has not received successive Keepalive and/or Update and/or Notification messages within the Hold Time period, then this error is communicated and the connection is closed.
   v) **Finite State Machine Error:** It includes any procedural error.
   vi) **Cease:** It is used by a router to close a connection with another router in the absence of any other error.

Ques 26) Discuss about the functional procedures of BGP.
Or
Explain how routing is done using BGP. (2018[05])

Ans: Functional Procedures of BGP/Routing in BGP
Three functional procedures are involved in BGP:

1) **Neighbour Acquisition:** Two routers are considered to be neighbours if they are attached to the same network. If the two routers are in different autonomous systems, they may wish to exchange routing information. For this purpose, it is necessary first to perform neighbour acquisition. In essence, neighbour acquisition occurs when two neighbouring routers in different autonomous systems agree to exchange routing information regularly.

2) **Neighbour Reachability:** Once a neighbour relationship is established, the neighbour reachability procedure is used to maintain the relationship. Each partner needs to be assured that the other partner still exists and is still engaged in the neighbour relationship. For this purpose, the two routers periodically issue Keepalive messages to each other.

3) **Network Reachability:** The final procedure specified by BGP is network reachability. Each router maintains a database of the networks that it can reach and the preferred route for reaching each network. When a change is made to this database, the router issues an Update message that is broadcast to all other routers implementing BGP. Because the Update message is broadcast, all BGP routers can build up and maintain their routing information.

Ques 27) Describe the format of IPv4 datagram with the help of a diagram, highlighting the significance of each field. (2018[06])

Or

Explain IPv4 with its datagram format?

Ans: IPv4 (Internet Protocol Version 4)

The Internet Protocol version 4 (IPv4) is the delivery mechanism used by the TCP/IP protocols.

IPv4 Datagram Format

Packets in the IPv4 layer are called datagrams. **Figure 4.15** shows the IPv4 datagram format.

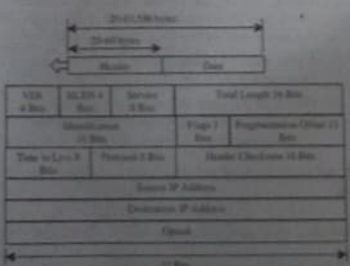


Figure 4.15: IPv4 Datagram Format

A datagram is a variable-length packet consisting of two parts: header and datainformation essential to routing and delivery. It is customary in TCP/IP to show the header in 4-byte sections. A brief description of each field is in order:

1) **Version (VER):** This 4-bit field defines the version of the IPv4 protocol. Currently the version is 4. However, version 6 (or IPng) may totally replace version 4 in the future.

2) **Header Length (HLEN):** This 4-bit field defines the total length of the datagram header in 4-byte words. This field is needed because the length of the header is variable (between 20 and 60 bytes).

3) **Services:** IETF has changed the interpretation and name of this 8-bit field. This field, previously called service type, is now called differentiated services.

4) **Total Length:** This is a 16-bit field that defines the total length (header plus data) of the IPv4 datagram in bytes. The header length can be found by multiplying the value in the HLEN field by 4.

Length of Data = Total Length – Header Length

5) **Identification:** This field is used in fragmentation.

6) **Flags:** This field is used in fragmentation.

7) **Fragmentation Offset:** This field is used in fragmentation.

8) **Time to Live:** This field was originally designed to hold a timestamp, which was decremented by each visited router.

9) **Protocol:** This 8-bit field defines the higher-level protocol that uses the services of the IPv4 layer. An IPv4 datagram can encapsulate data from several higher-level protocols such as TCP, UDP, ICMP and IGMP. This field specifies the final destination protocol to which the IPv4 datagram is delivered.

10) **Checksum:** The checksum in the IPv4 packet covers only to header and not data.

11) **Source Address:** This 32-bit field defines the IPv4 address of the source. This field must remain unchanged during the time the IPv4 datagram travels from the source host to the destination host.

12) **Destination Address:** This 32-bit field defines the IPv4 address of the destination. This field must remain unchanged during the time the IPv4 datagram travels from the source host to the destination host.

Ques 28) What is IPv6 addressing? Give the structure of IPv6 Packet?

Or

How many octets does the smallest possible IPv6 (IP version 6) datagram contain?

Ans: IPv6 (Internet Protocol Version 6) Addressing

IPv6 is backward compatible with and is designed to fix the shortcomings of IPv4, such as data security and maximum number of user addresses.

This is the next generation IP protocol. IPv6 increases the address space from 32 to 128 bits, providing for an unlimited (for all intents and purposes) number of networks and systems.

This increased size provides for a broader range of addressing hierarchies and a much larger number of addressable nodes.

Structure of IPv6

An IPv6 packet (**figure 4.16**) has the following general form:

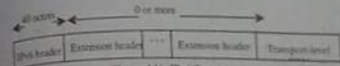


Figure 4.16: IPv6 Structure

The only header that is required is referred to simply as the IPv6 header. This is of fixed size with a length of 40 octets, compared to 20 octets for the mandatory portion of the IPv4 header. The following extension headers have been defined:

1) **Hop-by-Hop Options Header:** Defines special options that require hop-by-hop processing.

2) **Routing Header:** Provides extended routing, similar to IPv4 source routing.

3) **Fragment Header:** Contains fragmentation and reassembly information.

4) **Authentication Header:** Provides packet integrity and authentication.

5) **Encapsulating Security Pay-load Header:** Provides privacy.

6) **Destination Options Header:** Contains optional information to be examined by the destination node.

Ques 29) Draw and explain the datagram format for IPv6. (2018[05])

Or

Draw the IPv6 fixed header format. (2019[03])

Or

Explain IPV6 with frame format. (2020[05])

Ans: IPv6 Data Format/Fixed Header Format

IPv6 datagram format is shown in **figure 4.17**.

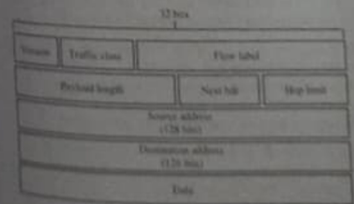


Figure 4.17

1) **Version:** This 4-bit field identifies the IP version number. Not surprisingly, IPv6 carries a value of 6 in this field. Note that putting a 4 in this field does not create a valid IPv4 datagram.

2) **Traffic Class:** This 8-bit field is similar in spirit to the TOS field we saw in IPv4.

3) **Flow Label:** This 20-bit field is used to identify a flow of datagrams.

4) **Payload Length:** This 6-bit value is treated as an unsigned integer giving the number of bytes in the IPv6 datagram following the fixed-length, 40-byte datagram header.

5) **Next Header:** This field identifies the protocol to which the contents (data field) of this datagram will be delivered (for example, to TCP or UDP). The filed uses the same values as the protocol field in the IPv4 header.

6) **Hop Limit:** The contents of this field are decremented by one by each router that forwards the datagram. If the hop limit count reaches zero, the datagram is discarded.

7) **Source and Destination Addresses:** The various formats of IPv6 128-bit addresses are described in RFC 4291.

8) **Data:** This is the payload portion of the IPv6 datagram. When the datagram reaches its destination, the payload will be removed from the IP datagram and passed on to the protocol specified in the next header field.

Ques 30) What are the different issues related to IPv6?

Or

Discuss about the issues with IPv6. (2019[03])

Ans: Issues Related to IPv6

The issues related to IPv6 network security are as follows:

1) **Lack of IPv6 Security Training/Education:** The biggest risk today is the lack of IPv6 security knowledge. Enterprises must invest time and money in IPv6 security training upfront, before deploying. Network security is more effective as part of the planning stage rather than after deployment.

2) **Security Device Bypass via Unfiltered IPv6 and Tunnelled Traffic:** Only a lack of knowledge is considered a bigger risk than the security products themselves. Conceptually it's simple, security products need to do two things – recognize suspicious IPv6 packets and apply controls when they do.

However in practice this is hardly possible in IPv4 let alone an environment that may have rogue or unknown tunnel traffic.

3) **Lack of IPv6 Support at ISPs and Vendors:** Thorough testing is critical until IPv6 security functionality and stability are on par with that of IPv4. A test network and a test plan for all protocols involved must be devised to test all equipment – especially new security tech from vendors. Every network is unique and requires a unique test plan. Further complicating the issue is not having a native IPv6 connection from provider. A tunnel connected to interface further increases security complexity and provides an opening for man-in-the-middle and denial-of-service attacks.

4) **Congruence of Security Policies in IPv4 & IPv6:** Weak IPv6 security policies are a direct result of the current deficit in IPv6 security knowledge. Not only do the depth of the IPv6 security policies need to be equal to that of their IPv4 counterparts but their breadth must be wider to encompass new vulnerabilities that didn't need to be considered in an IPv4 homogeneous environment.

Ques 31) Differentiate IPV4 and IPV6? (2020[05])

Or

What is difference between IPv6 and IPv4?

Ans: Difference between IPv6 & IPv4

Table below shows the difference between IPv6 and IPv4:

Table 4.2: Differentiate between IPv6 and IPv4

Description	IPv4	IPv6
Address	32 bits long (4 bytes). The text form of the IPv4 address is nnn.nnn.nnn.nnn, where 0<=nnn<=255, and each n is a decimal digit.	128 bits long (16 bytes). The text form of the IPv6 address is xxxx:xxxx:xxxx:xxxx: xxxx:xxxx:xxxx:xxxx, where each x is a hexadecimal digit, representing 4 bits.
Address Allocation	Originally, addresses were allocated by network class.	Allocatioc is in the earliest stages.
Address Lifetime	Generally, not an applicable concept.	IPv6 addresses have two lifetimes: preferred and valid, with the preferred lifetime always <= valid.
Address Mask	Used to designate network from host portion.	Not used
Address Types	Unicast, multicast, and broadcast.	Unicast, multicast, and anycast.
Domain Name System (DNS)	Applications accept host names and then use DNS to get an IP address, using socket API	Same for IPv6.
File Transfer Protocol (FTP)	File Transfer Protocol allows you to send and receive files across networks.	Some implementations of FTP does not support IPv6.
IP Header	Variable length of 20-60 bytes, depending on IP options present.	Fixed length of 40 bytes. There are no IP header options. Generally, the IPv6 header is simpler than the IPv4 header.

Ques 32) Write the ICMPv6 in details with advantages and disadvantages. (2020[07])

Or

What are the main functions of ICMPv6?

Ans: ICMPv6

Internet Control Message Protocol (both ICMPv4 and ICMPv6) is a protocol which acts as a communication messenger protocol between the communicating devices in IP network. ICMP messages provide feedback, error reporting and network diagnostic functions in IP networks which are necessary for the smooth operation of IPv6.

Internet Control Message Protocol Version 6 (ICMPv6) is a new version of the ICMP that forms an integral part of the Internet Protocol version 6 (IPv6) architecture. ICMPv6 messages are transported within IPv6 packet that may include IPv6 extension headers.

ICMPv6 is an integral part of IPv6 and ICMPv6 protocol in IPv6 and has much more importance and functions than ICMPv4 protocol in IPv4.

Functions of ICMPv6

Main functions of ICMPv6 are as follows:
1) Error Reporting
2) Network Diagnostics
3) Neighbor Discovery
4) Multicast Membership Reporting
5) Router Solicitation and Router Advertisements

Advantages of ICMPv6

1) If a wrong IP address is used for configuring a client to the DNS server, an ICMP message is sent by the destination device to indicate the error.
2) If a program does not allow fragmentation of its communications but it is required to communicate with a destination device, the router undertaking the fragmentation of the packet sends an ICMP message to the source device to indicate the error.
3) If a client sends all communications to a particular router despite another router offering a best route, the particular router responds with the IP address of the router that provides a better route in the form of an ICMP message.

Disadvantages of ICMPv6

1) Sending a lot of ICMP packets increases network traffic.
2) A device's performance degrades if it receives a lot of malicious packets that cause it to respond with ICMP error packets.
3) A host's performance degrades if the redirect function adds many routes to its routing table.
4) End users are affected if malicious users send many ICMP destination unreachable packets.

Ques 33) What is the packet format of ICMPv6? Also discuss the message types of ICMPv6.

Ans: Packets Format

ICMPv6 packets have the format shown in the **figure below**:

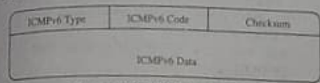


Figure 4.18: ICMPv6 Packet Format

The 8-bit Type field indicates the type of the message. If the high-order bit has value zero (values in the range from 0 to 127), it indicates an error message; if the high-order bit has value 1 (values in the range from 128 to 255), it indicates an information message. The 8-bit Code field content depends on the message type. Checksum field helps in the detection of errors in the ICMP message and in part of the IPv6 message.

ICMPv6 Message Types

ICMPv6 is a multipurpose protocol as it is used for a plethora of activities such as reporting errors encountered in processing data packets, reporting multicast memberships, performing Neighbor Discovery, and performing diagnostics. An ICMP message is identified by a value of 58 in the Next Header field of the IPv6 header or of the preceding Header. A list of currently defined message types is shown in the **table below**.

Table 4.3: ICMPv6 Message Types

Type	Meaning
1	Destination Unreachable
2	Packet Too Big
3	Time Exceeded
4	Parameter Problem
128	Echo Request
129	Echo Reply
130	Group Membership Query
131	Group Membership Report
132	Group Membership Reduction
133	Router Solicitation
134	Router Advertisement
135	Neighbor Solicitation
136	Neighbor Advertisement
137	Redirect
138	Router Renumbering

Ques 34) Describe the ICMPv6 message.

Or

Explain the error and Information message of ICMPv6.

Ans: ICMPv6 Messages

ICMPv6 is a multipurpose protocol and is used for a variety of activities including error reporting in packet processing, diagnostic activities, Neighbor Discovery process and IPv6 multicast membership reporting. To perform these activities, ICMPv6 messages are subdivided into two classes:

1) **Error Messages:** ICMPv6 error messages are used to report errors in the forwarding or delivery of IPv6 packets. The ICMPv6 "Type field" values for the error message are between 0 and 127. The Internet Control Message Protocol Version 6 (ICMPv6) error messages belong to four different categories:
   i) **Destination Unreachable:** Destination Unreachable ICMPv6 error message is generated by the source host or a router when an IPv6 datagram packet cannot be delivered for any reason other than congestion.
   ii) **Packet Too Big:** Packet Too Big ICMPv6 error messages are generated by the router when a packet cannot be forwarded to the next hop link because the size of the IPv6 datagram is larger than the MTU (Maximum Transmission Unit) of the link. Packet Too Big ICMPv6 error message includes the MTU of the next link also. MTU is the size of the largest protocol data unit that is supported over the link.
   iii) **Time Exceeded:** Similar to the Time-to-Live field value in IPv4 datagram header, IPv6 header includes a Hop Limit field. The Hop Limit field value in IPv6 header is used to prevent routing loops. Hop Limit field in IPv6 datagram header is decremented by each router that forwards the packet. When the Hop Limit field value in IPv6 header reaches zero, the router discards the IPv6 datagram packet and returns a "Time Exceeded" ICMPv6 error message to the source host.
   iv) **Parameter Problems:** Parameter Problem ICMPv6 error message is typically related with the problems and mistakes related with IPv6 header itself. When a problem or mistake with an IPv6 header make a router cannot process the packet, the router stops processing the IPv6 datagram packet, discards the packet and returns a "Parameter Problem" ICMPv6 error message to the source host.
2) **Information Messages:** ICMPv6 informational messages are used for network diagnostic functions and additional critical network functions like Neighbor Discovery, Router Solicitation & R…

Advertisements, Multicast Memberships, Echo Request and Echo Reply are also ICMPv6 informational messages. ICMPv6 informational messages have values for the Type field (8 bit binary number) between 128 and 255.

Internet Control Message Protocol Version 6 (ICMPv6) information messages are subdivided into three groups:

i) **Diagnostic Messages:** ICMPv6 Echo request and Echo reply are the Diagnostic messages. Every IPv6 host must return an ICMPv6 Echo reply when it receives an ICMPv6 Echo request. Echo request and Echo reply messages are used by the ping command to check the network connectivity between two IPv6 hosts.

ii) **MLD (Multicast Listener Discovery) Messages:** ICMPv6 MLD Messages are used by an IPv6 enabled router to discover hosts who are interested in multicast packets, and the multicast addresses they are interested. MLD messages are used by MLD Protocol. MLD (Multicast Listener Discovery) Protocol is the IPv6 equivalent of IGMP (Internet Group Management) Protocol in IPv4.

iii) **ND (Neighbor Discovery) Messages:** ICMPv6 ND Messages are used for the Neighbor Discovery Protocol (NDP). ND Messages includes Router Solicitation & Router Advertisement, Neighbor Solicitation and Neighbor Advertisement.

Module 5

Transport Layer and Application Layer

TRANSPORT LAYER

Ques 1) What is transport layer? What are the functions of transport layer?

Ans: Transport Layer

The transport layer is the fourth layer of the OSI reference model. Transport layer provides transparent, reliable, and cost effective transfer of data units between the upper layer entities in the end systems.

Functions of Transport Layer

The basic functions of Transport Layer are:

1) **End-to-End Delivery:** The network layer oversees the end-to-end delivery of individual packets but does not see any relationship between those packets, even those belonging to a single message. It treats each as an independent entity.

2) **Addressing:** The client needs the address of the remote computer it wants to communicate with. Such remote computers have a unique address so that it can be distinguished from all the other computers.

3) **Reliable Delivery:** The reliable delivery considers the following issues given below:
   i) Error Control
   ii) Sequence Control
   iii) Loss Control
   iv) Duplication Control

4) **Flow Control:** Fast host cannot keep pace with a slow one. Hence, this is a mechanism to regulate the flow of information. The amount of memory on a computer is limited, and without flow control a larger computer might flood a computer with so much information that it can't hold it all before dealing with it.

5) **Multiplexing:** To improve transmission efficiency, the transport layer has the option of multiplexing.

Ques 2) What is transport service? Also explain the service provided to upper layers.

Or

What do mean by transport service primitives.

Ans: Transport Service

The transport service is said to perform "peer to peer" communication, with the remote (peer) transport entity. The data communicated by the transport layer is encapsulated in a transport layer PDU and sent in a network layer SDU.

The network layer nodes (i.e., Intermediate Systems (IS)) transfer the transport PDU intact, without decoding or modifying the content of the PDU. In this way, only the peer transport entities actually communicate using the PDUs of the transport protocol.

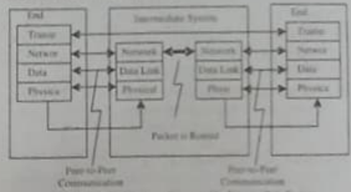


Figure 5.1: Two End Systems Connected by Intermediate System

The transport layer relieves the upper layers from any concern with providing reliable and cost effective data transfer. It provides end-to-end control and information transfer with the quality of service needed by the application program. It is the first true end-to-end layer, implemented in all End Systems (ES).

Service Provided to Upper Layers

To achieve this goal, the transport layer makes use of the services provided by the network layer. The hardware and/or software within the transport layer that does the work are called the **transport entity**.

The transport entity can be located in the operating system kernel, in a separate user process, in a library package bound into network applications, or conceivably on the network interface card. The (logical) relationship of the network, transport, and application layers is shown in **figure 5.2**.

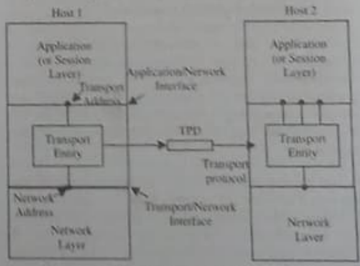


Figure 5.2: Network, Transport, and Application Layers

It primarily deals with:
1) Accepting APDU from the Application layer through the SAP.
2) Processing these APDU.
3) Deciding transport connection requirements (for further transmitting this DU after encapsulating it within a TPDU).
4) Passing this packet through the SAP to the lower layer (NL).
5) Accepting TPDU from the lower layer through the SAP.
6) Processing the TPDU.
7) Removing the encapsulation and passing the APDU through the SAP to the Upper layer (Application layer).
8) Providing support for connection-oriented/connectionless services as the case may be (depending upon the protocol stack and need).
9) Provide diagnostic support for network monitoring, configuration, management and troubleshooting at the Transport layer or higher layer.

Transport Service Primitives
Transport service allows application programs to establish, use, and then release connections, which is sufficient for many applications. The different transport service primitives are shown in **table 5.1**:

Table 5.1: Primitives for a Simple Transport Service

Primitive	Packet Sent	Meaning
LISTEN	(none)	Block until some process tries to connect
CONNECT	CONNECTION REQ	Actively attempt to establish a connection
SEND	DATA	Send information
RECEIVE	(none)	Block until a DATA packet arrives
DISCONNECT	DISCONNECTION REQ	This side wants to release the connection

...ues 3) What are port numbers, give its importance ... computer communication? **(2019[03])**

...s: Port Numbers and Its Importance
...ort number is the logical address of each application or ...ess that uses a network or the Internet to ...municate. A port number uniquely identifies a ...rk-based application on a computer. Each ...ation/program is allocated a 16-bit integer port ...r. This number is assigned automatically by the OS, ...lly by the user or is set as a default for some popular ...ions.

...umber primarily aids in the transmission of data ...a network and an application. Port numbers work ...ration with networking protocols to achieve this. ...**ple,** in an incoming message/packet, the IP ...used to identify the destination computer/node, ...port number further specifies the destination application/program in that computer. Similarly, all outgoing network packets contain application port numbers in the packet header to enable the receiver to distinguish the specific application.

Ques 4) What is TCP? Write the TCP packet format?
Or
Describe the format of a TCP segment with the help of a diagram. **(2021[05])**

Ans: Transmission Control Protocol (TCP)
The Transmission Control Protocol (TCP) is a connection-oriented reliable protocol. It provides a reliable transport service between pairs of processes executing on End Systems (ES) using the network layer service provided by the IP protocol. It is the general protocol suite of the Internet; encompassing protocols for network activities such as datagram delivery ad acknowledgement and protocols for user activity such as remote login (Telnet) and file transfer protocol (FTP).

The basic protocol used by TCP entities is the sliding window protocol. When a sender transmits a segment, it also starts a timer. When the segment arrives at the destination, the receiving TCP entity sends back a segment (with data if any exist, otherwise without data) bearing an acknowledgement number equal to the next sequence number it expects to receive. If the sender's timer goes-off before the acknowledgement is received, the sender transmits the segment again.

TCP Packet Format/TCP Segment Header
The following descriptions summarise the TCP packet fields illustrated in **figure 5.3**:

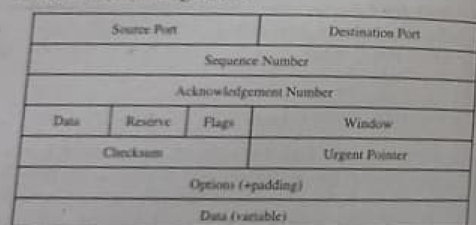


Figure 5.3: TCP Packet Format

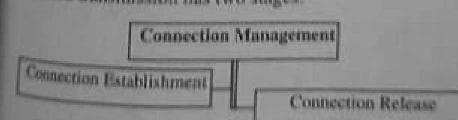
1) **Source Port and Destination Port:** Identifies points at which upper-layer source and destination processes receive TCP services.
2) **Sequence Number:** Usually specifies the number assigned to the first byte of data in the current message. In the connection-establishment phase, this field also can be used to identify an initial sequence number to be used in an upcoming transmission.
3) **Acknowledgment Number:** Contains the sequence number of the next byte of data the sender of the packet expects to receive.
4) **Data Offset:** Indicates the number of 32-bit words in the TCP header.

5) **Reserved:** Remains reserved for future use.
6) **Flags (6 bits):** For each flag, if set to 1, the meaning is as follows:
   i) **CWR:** Congestion window reduced.
   ii) **ECE:** ECN-Echo; the CWR and ECE bits, defined in RFC 3168, are used for the explicit congestion notification function.
   iii) **URG:** Urgent pointer field significant.
   iv) **ACK:** Acknowledgment field significant.
   v) **PSH:** Push function.
   vi) **RST:** Reset the connection.
   vii) **SYN:** Synchronise the sequence numbers.
   viii) **FIN:** No more data from sender.
7) **Window:** Specifies the size of the sender's receiving window (that is, the buffer space available for incoming data).
8) **Checksum:** Indicates whether the header was damaged in transit.
9) **Urgent Pointer:** Points to the first urgent data byte in the packet.
10) **Options:** Specifies various TCP options.
11) **Data:** Contains upper-layer information.

Ques 5) Discuss about the connection management modelling in detail.

Ans: Connection Management
The network layer oversees the end-to-end delivery of individual packets but does not see any relationship between those packets, even those belonging to a single message. It treats each as an independent entity. The transport layer, on the other hand, makes sure that the entire message (not just a single packet) arrives intact. Thus, it oversees the **end-to-end** (source to destination) of the entire message. End-to-end delivery can be accomplished in either of two modes:
1) Connection-oriented
2) Connectionless

Of these two, the connection-oriented mode is the more commonly used. A connection-oriented protocol establishes a virtual circuit or pathway through the internet between the sender and receiver. All of the packets belonging to a message are then sent over this same path. Using a single pathway for the entire message facilitates the acknowledgment process and retransmission of damaged or lost frames. Connection-oriented services, therefore, are generally considered reliable. Connection-oriented transmission has two stages:



Ques 6) Discuss how the connection is established and released in the TCP.

Ans: TCP Connection Establishment
To establish a connection, one side, say, the server passively waits for an incoming connection by executing the LISTEN and ACCEPTS primitives, either specifying a specific source or nobody in particular.

The other side, say, the client, executes a CONNECT primitive, specifying the IP address and port to which it wants to connect, the maximum TCP segment size it is willing to accept, and optionally some user data (e.g., a password).

The CONNECT primitive sends a TCP segment with the SYN bit on and ACK bit-off and waits for a response.

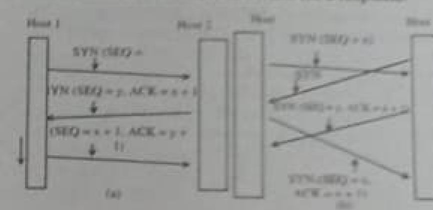


Figure 5.4: (a) TCP Connection Establishment in the Normal Case (b) Call Collision

When this segment arrives at the destination, the TCP entity there checks to see if there is a process that has done a LISTEN on the port given in the Destination port field.

If not, it sends a reply with the RST bit onto reject the connection. If some process is listening to the port, that process is given the incoming TCP segment.

It can then either accept or reject the connection. If it accepts, an acknowledgement segment is sent back. The sequence of TCP segments sent in the normal case is shown in **figure 5.4**.

In the event that two hosts simultaneously attempt to establish a connection between the same two sockets, the sequence of events is as illustrated in **figure 5.4(b)**.

The result of these events is that just one connection is established, not two because connections are identified by their end points.

TCP Connection Release
TCP connections are full duplex, now connections are released, it is best to think of them as a pair of simplex connections. Each simplex connection is released independently of its sibling.

To release a connection, either party can send a TCP segment with the FIN bit set, which means that it has no more data to transmit. When the FIN is acknowledged, that direction is shut-down for new data.

Data may continue to flow indefinitely in the other direction, however. When both directions have been shut down, the connection is released. Normally, four TCP segments are needed to release a connection, one FIN and one ACK for each direction.

However, it is possible for the first ACK and the second FIN to be contained in the same segment, reducing the total count to three.

To avoid the two-army problem, timers are used. If a response to a FIN is not forthcoming within two maximum packet lifetimes, the sender of the FIN releases the connection.

Ques 7) Discuss about the two-way and three-way handshake.

Or

Explain the three different phases in a TCP transmission with the help of diagrams. (2018[07])

Ans: Two-Way Handshake

When establishing a connection, transport entity must take into account of reliability or unreliability of network service. The problem occurs when the network loses, store and duplicate packets. Connection establishment is by mutual agreement from each transport sender and receiver entity and on different parameters. A two-way handshake connection establishment is shown in **figure 5.5.**

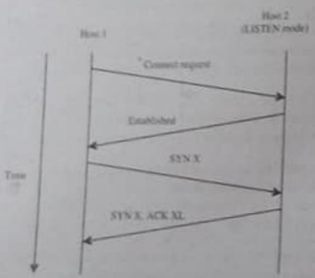


Figure 5.5: Connection Establishment using Two-way Handshake

It can be accomplished by a simple set of connection management primitives using two-way handshake.

If the destination transports entity in the LISTEN State for the port, then a connection is established through the following action by the receiving transport entity:

1) Signal the source transport entity that a connection is open.
2) Send a SYN (for synchronise) as conformation to the remote transport entity (i.e., receiving entity).
3) Put the connection is an established state (ESTAB).

Either side can initiate a connection, if both sides initiate a connection at the same time, it is established without confusion. The connection is prematurely terminated if either side issue a close command.

Problems with Two-Way Handshake

1) Two-way handshake suffers from duplicate or lost of SYN or ACK signal during connection establishment. Suppose host A issue an SYN to host B. It expects to get an SYN back, to confirming the connection. There may be possibilities of A's SYN lost or B's ACK lost. Both cases can be handled by use of a retransmit SYN timer. After host A issue an SYN, it will reissue the SYN when the timer expires but this rises to duplicate SYNs.

2) If B's ACK was lost, the B may receive two SYNs from A. If B's ACK was not lost but simply delayed, host A may get two ACK. In all this cases host A and B must ignore duplicate SYNs once a connection is established.

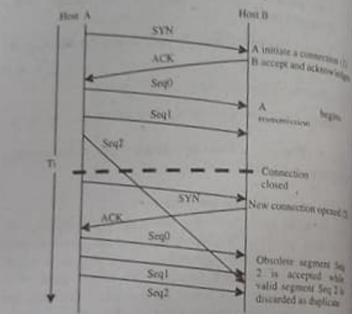


Figure 5.6: Two Way Handshake Problem

Problem in two-way handshake is illustrated in **figure 5.6.** Assume that with each new connection, each transport protocol entity begins numbering its data segment with sequence number 0.

3) Another problem with two-way handshake is that it suffers from obsolete SYN segments. **Figure 5.7** depicted this problem that may arise.

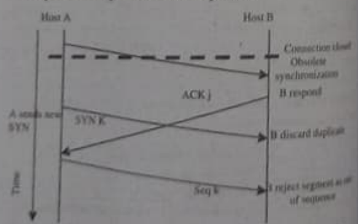


Figure 5.7: Two-way Handshake Suffers with Obsolete SYN Segments

An old connection request arrives (e.g. SYN) at host B after the connection is terminated. Host B assume this is a fresh request and responds with ACK, meanwhile, A has decided to open a new connection with B and sends SYN k. B discards this as a duplicate.

Now both sides have transmitted and subsequently received a SYN segment, and therefore think that a valid connection exits. However, when A initiate data transfer with numbered k, host B rejects the segment as being out of sequence.

Three-Way Handshake/ Three Different Phases in a TCP Transmission

To solve out these problems, each side to acknowledged explicitly the others connection request (i.e., SYN) and a sequence number, the procedure is known as three-way handshake.

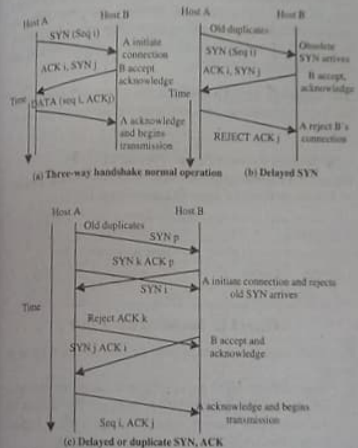


Figure 5.8: Three-way Handshake to Establishing a Connection

Figure 5.8 depicted typical three-way handshake connection establishment operations. This connection establishment protocol does not require both sides to begin sending with the same sequence number, so it can be used with synchronisation methods other than the global clock method.

Ques 8) Discuss about the TCP retransmission policy.

Ans: TCP Retransmission Policy

The TCP retransmission means resending the packets over the network that have been either lost or damaged. Here, retransmission is a mechanism used by protocols such as TCP to provide reliable communication. Here, reliable communication means that the protocol guarantees packet's delivery even if the data packet has been lost or damaged.

The networks are unreliable and do not guarantee the delay or the retransmission of the lost or damaged packets. The network which uses a combination of acknowledgment and retransmission of damaged or lost packets offers reliability.

Retransmission Mechanism

Here, retransmission means the data packets have been lost, which leads to a lack of acknowledgment. This lack of acknowledgment triggers a timer to timeout, which leads to the retransmission of data packets. Here, the timer means that if no acknowledgment is received before the timer expires, the data packet is retransmitted. Let's consider the following scenarios of retransmission.

Scenario 1: When the Data Packet is Lost or Erroneous: In this scenario, the packet is sent to the receiver, but no acknowledgment is received within that timeout period. When the timeout period expires, then the packet is resent again. When the packet is retransmitted, the acknowledgment is received. Once the acknowledgment is received, retransmission will not occur again(**figure 5.9**).

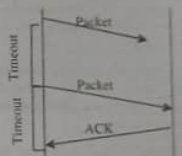


Figure 5.9: Packet Lost of Erroneous/Dropped at Rx

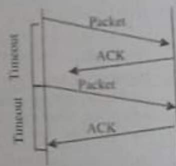


Figure 5.9: ACK Lost Duplicate Packet

Scenario 2: When the Packet is Received but the Acknowledgment is Lost: In this scenario, the packet is received on the other side, but the acknowledgment is lost, i.e., the ACK is not received on the sender side. Once the timeout period expires, the packet is resent. There are two copies of the packets on the other side; though the packet is received correctly, the acknowledgment is not received, so the sender retransmits the packet. In this case, the retransmission could have been avoided, but due to the loss of the ACK, the packet is retransmitted (**figure 5.10**).

Scenario 3: When the Early Timeout Occurs: In this scenario, the packet is sent, but due to the delay in acknowledgment or timeout has occurred before the actual timeout, the packet is retransmitted. In this case, the

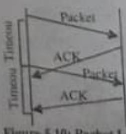


Figure 5.10: Packet Lost of Erroneous/Dropped at Rx

packet has been sent again unnecessarily due to the delay in acknowledgment or the timeout has been set earlier than the actual timeout.

In the above scenarios, the first scenario cannot be avoided, but the other two scenarios can be avoided. Let's see how we can avoid these situations. The sender sets the timeout period for an ACK. The timeout period can be of two types:

1) **Too short:** If the timeout period is too short, then the retransmissions will be wasted.
2) **Too long:** If the timeout period is too long, then there will be an excessive delay when the packet is lost.

In order to overcome the above two situations, TCP sets the timeout as a function of the RTT (round trip time) where round trip time is the time required for the packet to travel from the source to the destination and then come back again.

Ques 9) What are the different TCP services?

Ans: TCP Services

There is a long list of services that can be optionally provided by the Transport Layer.

All available services are:

1) **Connection-Oriented:** This is normally easier to deal with than connection-less models, so where the Network layer only provides a connection-less service, often a connection-oriented service is built on top of that in the Transport Layer.
2) **Reliable Data:** Packets may be lost in routers, switches, bridges and hosts due to network congestion, when the packet queues are filled and the network nodes have to delete packets.

   Packets may be lost or corrupted in Ethernet due to interference and noise, since Ethernet does not retransmit corrupted packets. Packets may be delivered in the wrong order by an underlying network.

   By means of an error detection code, **for example** a checksum, the transport protocol may check that the data is not corrupted, and verify that by sending an ACK message to the sender. Automatic repeat request schemes may be used to retransmit lost or corrupted data.
3) **Flow Control:** The amount of memory on a computer is limited, and without flow control a larger computer might flood a computer with so much information that it can't hold it all before dealing with it. Flow control allows the receiver to respond before it is overwhelmed.
4) **Congestion Avoidance:** Network congestion occurs when a queue buffer of a network node is full and starts to drop packets. Automatic repeat request may keep the network in a congested state.

   This situation can be avoided by adding congestion avoidance to the flow control, including slow-start.
5) **Ports:** Ports are essentially ways to address multiple entities in the same location. **For example,** the first line of a postal address is a kind of port, and distinguishes between different occupants of the same house.

Ques 10) What is UDP? Also explain the UDP header format.

Or

Describe the operation and packet format of UDP. (2018[05])

Or

Explain the different operations performed by UDP.

Or

Define the role of UDP in Internet protocol suite. (2021[05])

Ans: User Datagram Protocol (UDP)

User Datagram Protocol (UDP) provides a minimal, unreliable, best-effort, message-passing transport to applications and upper-layer protocols. Service provided by UDP is an unreliable service that provides no guarantees for delivery and no protection from duplication (e.g. if this arises due to software errors within an Intermediate System (IS)).

The simplicity of UDP reduces the overhead from using the protocol and the services may be adequate in many cases.

UDP Header Format/Packet Format of UDP

Figure 5.11 shows that the UDP protocol header consists of 8 bytes of Protocol Control Information (PCI).

Bits	0-15	16-31
0	Source port	Destination port
32	Length	Checksum
64	Data	

Figure 5.11: Header Format of UDP

The UDP header consists of four fields each of 2 bytes in length:

1) **Source Port:** UDP packets from a client use this as a service access point (SAP) to indicate the session on the local client that originated the packet. UDP packets from a server carry the server SAP in this field.
2) **Destination Port:** UDP packets from a client use this as a service access point (SAP) to indicate the service required from the remote server. UDP packets from a server carry the client SAP in this field.
3) **UDP Length:** The number of bytes comprising the combined UDP header information and payload data.
4) **UDP Checksum:** A checksum to verify that the end to end data has not been corrupted by routers or bridges in the network or by the processing in an end system. The algorithm to compute the checksum is the Standard Internet Checksum algorithm.

UDP Operations

The operations performed by UDP are as below:

1) **Connectionless Service**
   i) This means that each user datagram sent by UDP is an independent datagram i.e., no relationship between the user datagrams even if they belong to the same destination program.
   ii) The user datagrams are not numbered; there is no connection establishment and no connection release.
   iii) This means that each user datagram can travel on a different path.
   iv) A process using UDP cannot send a stream of data. Each request should be small enough to fit into one user datagram. Only those processes sending short messages should use UDP.
2) **Flow and Error Control**
   i) There is no flow control. The receiver may then overflow.
   ii) There is no error control except for the checksum. The sender could not know if the message has been lost or duplicated. The receiver silently discards a user datagram when an error is detected by the checksum.
   iii) The process using UDP should provide the flow and error control if they are needed.
   iv) No connection state (sequence and ACK numbers, send and receive buffers? etc.) is needed.
3) **Encapsulation and Decapsulation:** To send a message from one process to another, the UDP protocol encapsulates and decapsulates messages in an IP datagram. In UDP, queues are associated with ports. At the client site, when a process starts, it requests a port number from the operating system.

   When a message arrives for a client, UDP checks to see if an incoming queue has been created for the port number specified in the destination port number field of the user datagram. If there is such a queue, UDP sends the received user datagram to the end of the queue.
4) **Queuing:** In UDP, queues are associated with ports.
   i) **At Client Site**
      a) When a process starts, it requests a port number from the OS.
      b) Some implementations create both incoming and outgoing queue associated with each process. Other implementations create only an incoming queue.
      c) These queues are identified by the ephemeral port numbers assigned. These queues function as long as the process is running. They are destroyed when the process terminates.
      d) The client process can send messages to the outgoing queue by using the source port number specified in the request.
      e) An outgoing queue can overflow. The OS asks then the client to wait before sending any more messages.
      f) When a message arrives for a client, UDP checks if an incoming queue has been created for the port number specified in the destination port. If so, UDP sends the received user datagram to the end of the queue. Otherwise, UDP discards the user datagram and asks ICMP to send a port unreachable message to the server.
      g) An incoming queue can overflow. UDP drops then the user datagram and asks for a port unreachable message to be sent to the server.
   ii) **At Server Site**
      a) The mechanism of creating queues is different.
      b) The server asks for incoming and outgoing queues, using its well-known ports, when it starts. These queues remain open as long as the server is running.
      c) When a message arrives to the server, UDP checks to if an incoming queue has been created for the port number specified in the destination port number.

         If so, UDP places the user datagram at the end of the queue. Otherwise, UDP discards the user datagram and asks ICMP to send an unreachable port message to the client.
      d) An incoming queue can overflow. UDP drops the user datagram and asks that a port unreachable message to be sent to the client.
      e) When a server wants to respond to a client, it sends messages to the outgoing queue using the source port number specified in the request. UDP encapsulates the user datagram get from the outgoing queue in IP packets.
      f) If the outgoing queue overflows, the OS asks the server to wait before sending any more messages.

Ques 11) Explain the procedure for calculating the UDP checksum? (2019[03])

Ans: Procedure to Calculate UDP Checksum

Refer Module-6, Page No. D-92, Question No.7

UDP Checksum calculation is similar to TCP Checksum computation. It's also a 16-bit field of one's complement of one's complement sum of a pseudo UDP header + UDP datagram:

1) **Sender Side**
   i) It treats segment contents as sequence of 16-bit integers.
   ii) All segments are added. Let's call it sum.
   iii) **Checksum:** 1's complement of sum.(In 1's complement all 0s are converted into 1s and all 1s are converted into 0s).

iv) Sender puts this checksum value in UDP checksum field.

2) **Receiver Side**
   i) Calculate checksum
   ii) All segments are added and then sum is added with sender's checksum.
   iii) Check that any 0 bit is presented in checksum. If receiver side checksum contains any 0 then error is detected. So the packet is discarded by receiver.

Ques 12) What is difference between UDP and TCP?

Or

Distinguish between TCP and UDP header format. (2019[07])

Ans: Difference between UDP and TCP

Table 5.2 shows the major difference between UDP and TCP protocols:

Table 5.2: Comparison between UDP and TCP

Characteristics	UDP	TCP
General Description	Simple, high speed, low functionality "wrapper" that interfaces applications to the network layer and does little else.	Full-featured protocol that allows applications to send data reliable without worrying about network layer issues.
Data Interface to Application	Message-based; data is sent in discrete packages by the application.	Stream-based; data is sent by the application with no particular structure.
Reliability and Acknowledgments	Unreliable, best-effort delivery without acknowledgements	Reliable delivery of messages; all data is acknowledged.
Applications and Protocols	Multimedia applications, DNS,BOOTP,DHCP,TFTP, SNMP,RIP,NFS	FTP, Telnet, SMTP, DNS,HTTP POP,NNTP,IMAP,BGP, IRC,NFS

Ques 13) Discuss the TCP congestion control and different congestions detecting methods.

Ans: Congestion Control

Congestion control concerns controlling traffic entry into a telecommunications network, so as to avoid congestive collapse by attempting to avoid over-subscription of any of the processing or link capabilities of the intermediate nodes and networks and taking resource reducing steps, such as reducing the rate of sending packets.

Detecting Congestions

Detection in the internet is done in two different ways, which are:

1) **Implicit Method:** Whenever there is a segment loss, the TCP assumes that there is congestion. It sends application data in form of segments. It also sets the timer for each segment's ACK to come back.

   Whenever the TCP finds that an ACK does not come back when the timer times out, it concludes that there is congestion and takes remedial action.

2) **Explicit Method:** This method is recently added to TCP. In this method, two bits in the TCP header as well as the IP header are set aside to indicate congestion and as soon as the router realises that there is a likelihood of congestion, it sets one of the IP header bits in those going in the reverse direction of congestion as an indicator. The receiver, upon receiving a segment with the specified bit turned on, understands that congestion is building up in the area from where the segment has arrived. This is known as **Explicit Congestion Notification** (ECN). After realising about the congestion, the receiver relays the news to the sender using the TCP header bits.

Ques 14) Discuss the different congestion control algorithms.

Or

Write short notes on the following:

1) **Slow Start**
2) **Congestion Avoidance**
3) **Fast Retransmit**
4) **Fast Recovery**

Ans: Congestion Control Algorithms

The congestion control algorithms are:

1) **Slow Start:** The slow start algorithm regulates the flow of datagrams in the network to avoid congestion. With slow start algorithm, TCP monitors to make sure that the rate new packets are sent over the network are the rate at which the acknowledgements are returned by the receiver.

   Algorithm of Slow Start

   Step 1) Add a congestion window, cwnd, to the per-connection state. When starting or re-starting after a loss, set cwnd to one packet.

   Step 2) On each Ack for new data, increase cwnd by one packet.

   Step 3) When sending, send the minimum of the receiver's advertised window and cwnd.

2) **Congestion Avoidance:** During the initial data transfer phase of a TCP connection the Slow Start algorithm is used. However, there may be a point during Slow Start that the network is forced to drop one or more packets due to overload or congestion. If this happens, Congestion Avoidance is used to slow the transmission rate. However, Slow Start is used in conjunction with Congestion Avoidance as

the means to get the data transfer going again so it does not slow down and stay slow.

In the Congestion Avoidance algorithm a retransmission timer expiring or the reception of duplicate ACKs can implicitly signal the sender that a network congestion situation is occurring.

The sender immediately sets its transmission window to one half of the current window size (the minimum of the congestion window and the receiver's advertised window size), but to atleast two segments.

If congestion was indicated by a timeout, the congestion window is reset to one segment, which automatically puts the sender into Slow Start mode. If congestion was indicated by duplicate ACKs, the Fast Retransmit and Fast Recovery algorithms are invoked.

As data is received during Congestion Avoidance, the congestion window is increased. However, Slow Start is only used upto the halfway point where congestion originally occurred. This halfway point was recorded earlier as the new transmission window.

After this halfway point, the congestion window is increased by one segment for all segments in the transmission window that are acknowledged. This mechanism will force the sender to more slowly grow its transmission rate, as it will approach the point where congestion had previously been detected.

3) **Fast Retransmit:** Fast recovery and fast retransmit are based on the **Reno version of TCP**. TCP may generate an immediate duplicate Acknowledgment when segment are received in out of order manner.

   The purpose of this duplication is to let the sender know that segments sent are received out of order and to inform the sender what sequence number is expected in the next transmission.

   One knows that TCP does not know whether the duplicate Ack received is caused by a lost of segment or due to reordering of segment. TCP waits until small number duplicate Ack is received. It is assumed that when the problem is about the reordering of segments only two ACKs are sent from the receiver, before the reordered segment are processed and generate a new Ack.

   When three or more duplicate are received is a clear indication that the segment is lost. In this case, TCP will then perform a retransmission of the missing segment without waiting for the timer expiration.

4) **Fast Recovery:** In **Reno**, congestion avoidance is performed after fast retransmit sends the missing segment, since the lost packet is an indication of possible congestion. This algorithm is called **fast recovery**.

   This algorithm has worked remarkably well and believed to have prevented lot congestion on the internet. In the implementation of this algorithm, slow start is not performed.

   The reason for not performing slow start is to avoid reducing the flow between the two endpoints abruptly as there are still indications of communication. Since receiver can only generate an Ack when a segments is received, this confirms that the segment sent arrive at the receiver buffer, so slow start will not be necessary in this case.

   Fast recovery is believed to be an improvement which has allowed high throughput under reasonable congestions and scaled six orders of magnitude in size, speed, load and connectivity. It has also been relatively efficient at large windows.

APPLICATION LAYER

Ques 15) What is application layer? What are the functions of application layers?

Or

Explain the application layer of OSI reference model in detail with examples of protocols. (2021[04])

Ans: Application Layer

The application layer is the OSI layer closest to the end user, which means that both the OSI application layer and the user interact directly with the software application.

Some **examples** of application layer implementations include Telnet, File Transfer Protocol (FTP), and Simple Mail Transfer Protocol (SMTP).

Functions of Application Layers

1) **File Transfer, Access, and Management (FTAM):** This application allows a user to access files in a remote computer (to make changes or read data), to retrieve files from a remote computer; and to manage or control files in a remote computer.

2) **Addressing:** For communication between client and server there is requirement of address. When the client request is made to server, it also contains server address and its own address.

   The server response also contains destination address, i.e., the address of the client. DNS is used for this addressing.

3) **Mail Services:** This application provides the basis for e-mail forwarding and storage.

4) **Directory Services:** This application provides distributed database sources and access for global information about various objects and services.

5) **Authentication:** Authenticates the sender or receiver of the message or both.

Ques 16) What is the File Transfer Protocol (FTP)? What are the objectives of FTP?
Or
Explain the File Transfer Protocol (FTP) and its features. (2018[05])

Ans: File Transfer Protocol (FTP)
FTP allows the transfer of files from one computer to another. File can be in any format like text, graphics, sound, etc. It activates the client-server relationship. Thus, whatever that can be stored in a computer can be moved with the FTP service.

Objectives of FTP
1) Its main objective is to help in sharing of programs and data.
2) Inspiring the implicit use of remote computers is another objective of FTP.
3) To protect the user from variation in file storage systems among numerous hosts.
4) Effective and reliable sharing of data.

Features of FTP
1) FTP operates in a client/server environment, meaning that the remote machine is configured as a server, and consequently waits for the other machine (client) to request a service from it.
2) In UNIX, the service is provided by what is called a daemon, a small task that runs in the background. The FTP daemon is called ftpd.
3) The FTP protocol is used for transferring one file at a time, in either direction, between the client machine (the one which initiated the connection, i.e., the calling machine) and the server machine.
4) The FTP protocol can also perform other actions, such as creating and deleting directories (only if they are empty), listing files, deleting and renaming files, etc.
5) FTP allows files to have ownership and access restrictions.
6) FTP hides the details of individual computer systems.

Ques 17) Discuss about the mechanism of FTP.
Or
How FTP handles file transfer? (2019[03])

Ans: Mechanism of FTP
Figure 6.10 shows the mechanism of FTP. Its process of transferring a file is as follows:

1) First, define the address of remote computer on your computer as a parameter.
2) Then run the FTP command on your computer known as 'FTP client process', which makes a connection with the FTP process running on remote computer known as 'FTP server process'.
3) After running the FTP command user needs to enter the username and password to ensure that user is authorised to access the remote computer.
4) On successful login, the user is able to download or upload files using 'get' and 'put' commands. Listing of directories and navigating between directories before any transferred decision can also be done.

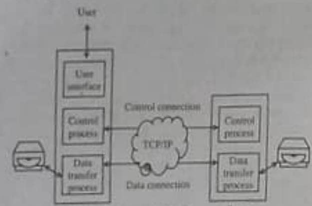


Figure 5.12: FTP

Ques 18) What is Domain Name Space (DNS)? What is the format of domain names? List out the different elements of DNS.

Ans: Domain Name Space (DNS)
Domain Name System (DNS) is a directory lookup service that provides a mapping between the name of a host on the Internet and its numerical address. DNS is essential to the functioning of the Internet.

Domain Name System (DNS) is service on a TCP/IP network which allows users of networks to utilise user-friendly names when looking for other hosts (i.e., computers) instead of having to remember and use their IP Addresses.

Format of Domain Names
Figure 5.13 shows a basic format of the domain name:

Figure 5.13: Basic Format of Domain Name

Types of Domain Name
There are two types of domain name:
1) Fully Qualified Domain Name(FQDN), and
2) Partially Qualified Domain Name(PQDN)

Elements of DNS
Four elements comprise the DNS are:
1) DNS Name Space
2) DNS Database
3) Name Servers
4) DNS Resolvers

Ques 19) Defined fully qualified and partially qualified domain name. (2021[03])

Ans: Fully Qualified Domain
A fully qualified domain name is distinguished by its lack of ambiguity. In detail, it can be interpreted only in one way. Usually, the FQDN consists of a hostname and at least one higher-level domain label. A fully qualified domain name consists of a list of domain labels representing the hierarchy from the lowest relevant level in the DNS to the TLD. The domain labels are separated by the full stop "."

For example, tppl.org.com explicitly specifies an absolute domain name that ends with the empty top-level domain label. A device with the hostname "tppl" in the parent domain minitool.com has the FQDN tppl.org.com. The FQDN uniquely distinguishes the device from any other hosts called "tppl" in other domains.

Partially Qualified Domain Name
A Partially Qualified Domain Name (PQDN) is used to specify a portion of a domain name, normally the host portion of it. A Partially Qualified Domain Name (PQDN) starts with a host name, but it may not reach up to the root.

By definition, a PQDN is ambiguous, because it does not give the full path to the domain. Thus, one can use a PQDN only within the context of a particular parent domain, whose absolute domain name is known. It is also called a relative domain name. PQDNs are usually simply hostnames, such as the left-most label in a fully qualified domain name.

A PQDN starts from a node, but it does not reach the root. It is used when the name to be resolved belongs to the same site as the client. Here resolver can supply the missing part, called the suffix, to create an FQDN.

Ques 20) Distinguish between partially qualified and fully qualified domain names. (2018[03])

Ans: Difference between Partially Qualified and Fully Qualified Domain Names

Partially Qualified Domain Names	Fully Qualified Domain Names
If a label is not terminated by a null string, it is called a Partially Qualified Domain Name (PQDN).	If a label is terminated by a null string, it is called a Fully Qualified Domain Name (FQDN).
A PQDN starts from a node, but it does not reach the root. It is used when the name to be resolved belongs to the same site, as the client.	A FQDN is a domain name that contains the full name of a host.
For example, the domain name **challenger**.	**For example,** the domain name challenger.atc.fhda.edu

Ques 21) Explain DNS message types. (2019[04])

Ans: DNS message types
DNS has two types of messages: query and response. Both types have the same format. The query message consists of a header and question records; the response message consists of a header, question records, answer records, authoritative records, and additional records (see **figure 5.14**).

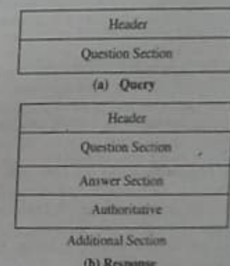


Figure 5.14: Query and Response Messages

The different sections are as follows:
1) **Header:** Both query and response messages have the same header format with some fields set to zero for the query messages. The header is 12 bytes and its format is shown in **figure 5.15**.

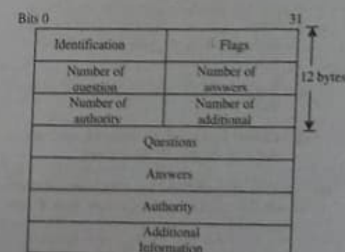


Figure 5.15: General Format of DNS

i) **Identification:** This is a 16-bit field used by the client to match the response with the query. The client uses a different identification number each time it sends a query. The server duplicates this number in the corresponding response.

ii) **Flags:** This is a 16-bit field consisting of the subfields shown in **figure 5.16**.

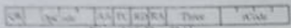


Figure 5.16: Flags Field

A brief description of each flag subfield follows:

a) **QR (Query/Response):** This is a 1-bit subfield that defines the type of message. If it is 0, the message is a query. If it is 1, the message is a response.

b) **OpCode:** This is a 4-bit subfield that defines the type of query or response (0 if standard, 1 if inverse, and 2 if a server status request).

c) **AA (Authoritative Answer):** This is a 1-bit subfield. When it is set (value of 1)it means that the name server is an authoritative server. It is used only in a response message.

d) **TC (Truncated):** This is a 1-bit subfield. When it is set (value of 1), it means that the response was more than 512 bytes and truncated to 512. It is used when DNS uses the services of UDP (see Section 19.8 on Encapsulation).

e) **RD (Recursion Desired):** This is a 1-bit subfield. When it is set (value of 1) it means the client desires a recursive answer. It is set in the query message and repeated in the response message.

f) **RA (Recursion Available):** This is a 1-bit subfield. When it is set in the response, it means that a recursive response is available. It is set only in the response message.

g) **Reserved:** This is a 3-bit subfield set to 000.

h) **rCode:** This is a 4-bit field that shows the status of the error in the response.

iii) **Number of Question:** This record contains the number of queries in the question section of the message.

iv) **Number of Answer:** This record contains the number of answer records in the answer section of the response message.

v) **Number of Authority:** This record contains the number of authority records in the authoritative section of the response message.

vi) **Number of Additional Records:** This record contains the number of authority records in the authoritative section of the response message.

2) **Question Section:** This is a section consisting of one or more question records. It is present on both query and response messages.

3) **Answer Section:** This is a section consisting of one or more resource records. It is present only on response messages. This section includes the answer from the server to the client (resolver).

4) **Authoritative Section:** This is a section consisting of one or more resource records. It is present only on response messages. This section gives information (domain name) about one or more authoritative servers for the query.

5) **Additional Information Section:** This is a section consisting of one or more resource records. It is present only on response messages. This section provides additional information that may help the resolver.

Ques 22) What is DNS Resolver? Also discuss about the recursive and iterative query.

Or

Explain about the address resolution mechanism.

Or

Describe the name-address resolution techniques used in DNS. (2018[05])

Ans: DNS Resolvers

The client-side of the DNS is called a DNS resolver. It is responsible for initiating and sequencing the queries that ultimately lead to a full resolution (translation) of the resource sought, e.g., translation of a domain name into an IP address.

DNS clients are configured with the addresses of DNS servers. Usually, these are servers which are authoritative for the domain of which they are a member. All requests for name resolution start with a request to one of these local servers.

DNS queries can be of two forms:

1) **Recursive Query:** A recursive query asks the nameserver to resolve a name completely, and return the result.

If the request cannot be satisfied directly, the nameserver looks in its configuration and caches for a server higher up the domain tree which may have more information. In the worst case, this will be a list of pre-configured servers for the root domain. These addresses are returned in a response called a referral. The local nameserver must then send its request to one of these servers.

2) **Iterative Query:** It asks the second nameserver to either respond with an authoritative reply, or with

the addresses of nameservers (NS records) listed in its tables or caches as authoritative for the relevant zone. The local nameserver then makes iterative queries, walking the tree downwards until an authoritative answer is found (either positive or negative) and returned to the client.

Process: Address Resolution Mechanism

Step 1: The local system is pre-configured with the known addresses of the root servers in a file of root hints, which need to be updated periodically by the local administrator from a reliable source to be kept up to date with the changes which occur over time.

Step 2: Query one of the root servers to find the server authoritative for the next level down (so in the case of our simple hostname, a root server would be asked for the address of a server with detailed knowledge of the example top level domain).

Step 3: Querying this second server for the address of a DNS server with detailed knowledge of the second-level domain (inadomain.example).

Step 4: Repeating the previous step to progress down the name, until the final step which would, rather than generating the address of the next DNS server, return the final address sought.

Ques 23) What is E-mail? What are the different e-mail protocols? List them.

Or

Discuss the working of electronic mail system. (2021[04])

Ans: Electronic Mail (E-Mail)

Electronic mail or e-mail, as it is popularly known, is a method of sending and receiving messages (mail) electronically over a computer network. E-mail is a system allows a person or a group to electronically communicate to others through Internet.

Figure below shows the working of email.

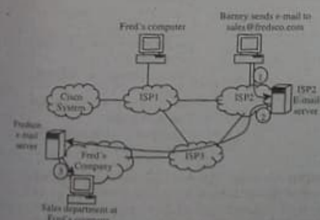


Figure 5.17 E-Mail in the Internet using Mail Servers

E-Mail Protocols

There are two main **protocols** used in E-Mails:

1) Simple Mail Transfer Protocol (SMTP)
2) Multipurpose Internet Mail Extension (MIME)

Ques 24) What is SMTP? Also explain the working of SMTP.

Or

What is the role of SMTP in E-Mail message transfer? (2019[03])

Ans: Simple Mail Transfer Protocol (SMTP)

It is a set of communication guidelines that allow software to transmit email over the Internet. Most email software is designed to use SMTP for communication purposes when sending email and it only works for outgoing messages.

When people set up their email programs, they will typically have to give the address of their Internet service provider's SMTP server for outgoing mail.

This protocol is used for the delivery of e-mail. When an E mail is to be sent, then the Mail Transfer Program contacts the remote machine and forms a TCP connection over which the mail is transferred.

Once the connection is established, then Simple Mail Transfer Protocol (SMTP) identifies the sender itself, specifies the recipient of mail and then transfers the E mail message.

Working of SMTP/Role of SMTP in E-Mail Transfer

Step 1) Composition of Mail: A user sending an e-mail starts by composing an electronic mail message using an authenticated mail client (Mail User Agent – MUA).

The message contains the body and the header. The body is the main part of the message while the header contains control information like the sender and recipient e-mail addresses.

Headers also include descriptive information like the subject and message submission date/time stamp. This is analogous to real mail, where the message body is like a letter and the header is like the envelope containing the recipient's address and a return address.

Step 2) Submission of Mail: The mail client then submits the completed e-mail to the configured SMTP server or mail server (Mail Submission Agent – MSA) using SMTP on TCP port 25 or 587, which acts as an electronic post-office. This is similar to how letters gets dropped off at the post-office for sorting and delivery.

Step 3) Delivery of Mail: E-mail addresses like john@email.com are sorted in a similar way. The "john" portion in the address is the username of the recipient and "email.com" is the domain name, similar to a postal address. If the domain name of the recipient's e-mail address is different from the sender, MSA will hand the mail over to the Mail Transfer Agent (MTA).

To relay the e-mail, the MTA must first locate the target domain. Once the record is located, MTA connects to the exchange server to relay the message.

Step 4) Receipt and Processing of Mail: Once the incoming message is accepted, the exchange server delivers it to the incoming mail server (Mail Delivery Agent MDA) which stores the e-mail where it waits for the user to retrieve it. This is equivalent to the real world example where the recipient's local post office delivers the mail into an individual's post office boxes.

Step 5) Access and Retrieval of Mail: The stored e-mail can be retrieved by authenticated mail clients (MUAs). By using a login and password to access the MUA, MDA ensures individual users only have the right to access their own e-mails.

Instead of SMTP, e-mail clients use either Internet Message Access Protocol (IMAP) or Post-Office Protocol (POP) to retrieve e-mails. POP is used for retrieving e-mails while IMAP manages and facilitate access to mail. Unlike SMTP, POP and IMAP are specifically designed to retrieve messages.

Ques 25) Discuss about the MIME?

Or

Write notes on MIME. **(2018[05])**

Or

Explain various features of MIME? **(2019[04])**

Ans : Multipurpose Internet Mail Extension (MIME)

The MIME specification includes the following **elements:**

1) Five new message **header fields** are defined. These fields provide information about the body of the message.

2) A number of content formats are defined, thus standardising representations that support multimedia electronic mail.

3) Transfer encodings are defined that enable the conversion of any content for mat into a form that is protected from alteration by the mail system.

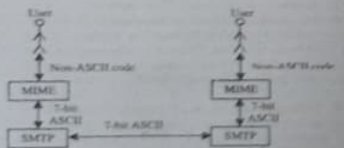


Figure 5.18: Working of MIME

Features of MIME

1) It is able to send multiple attachments with a single message.

2) Unlimited message length.

3) Binary attachments (Executables, images, audio, or video files) which may be divided if needed.

4) MIME provided support for varying content types and multi-part messages.

Header Fields

1) **MIME-Version:** It identifies the MIME version. These simply tell the user agent receiving the message that it is dealing with a MIME message, and which version of MIME it uses. Any message not containing a MIME-Version: header is assumed to be an English plaintext message, and is processed as such.

2) **Content-Description:** It is a human-readable string what is in the message. Header is an ASCII string telling what is in the message. This header is needed so the recipient will know whether it is worth decoding and reading the message.

3) **Content - ID:** It is a unique identifier. Header identifies the content. It uses the same format as the standard message-Id: header.

Type	Subtype	Description
Text	Plain	Unformatted text
	Richtext	Text including simple formatting commands
Image	GIF	Still picture in GIF format
	JPEG	Still picture in JPEG format
Audio	Basic	Audible sound
Video	MPEG	Movie in MPEG format
Application	Octet-stream	An un-interpreted byte sequence
	Postscript	A printable document in PostScript
Message	RFC822	A MIME RFC 822 message
	Partial	Message has been split for transmission
	External-body	Message itself must be fetched over the net
Multipart	Mixed	Independent parts in the specified order
	Alternative	Same message in different formats
	Parallel	Parts must be viewed simultaneously
	Digest	Each part is a complete RFC 822 message

4) **MIME Content Types:** There are seven different major types of content and a total of 15 subtypes. In general, a content type declares the general type of data, and the subtype specifies a particular format for that type of data.

The application type is a catchall for formats that require external processing not covered by one of the other types. The message type allows one message to be fully encapsulated inside another. This scheme is useful for forwarding e-mail.

The final type is multipart, which allows a message to contain more than one part, with the beginning and end of each part being clearly delimited.

5) **Content – Transfer – Encoding:** Tells how the body is wrapped for transmission through a network that may object to most characters other than letters, numbers, and punctuation marks.

Five schemes (plus an escape to new schemes) are provided. The simplest scheme is just ASCII text. The objective is to provide reliable delivery across the largest range of environments.

The Content-Transfer-Encoding field can actually take on six values, as listed in **table 6.3.** However, three of these values (7bit, 8bit and binary) indicate that no encoding has been done but provide some information about the nature of the data.

For SMTP transfer, it is safe to use the 7bit form. The 8bit and binary forms binary forms may be usable in other mail transport contexts. Another Content-Transfer-Encoding value is x-token, which indicates that some other encoding scheme is used, for which a name is to be supplied.

This could be a vendor-specific or application-specific scheme. The two actual encoding schemes defined are quoted-printable and base64.

Two schemes are defined to provide a choice between a transfer technique that is essentially human readable and one that is safe for all types of data in a way that is reasonably compact.

Table 5.3: MIME Content – Transfer Encoding

7bit	The data are all represented by short lines of ASCII characters.
8bit	The lines are short, but there may be non-ASCII characters (octets with the high-order bit set).
binary	Not only may non-ASCII characters be present, but the lines are not necessarily short enough for SMTP transport.
quoted-printable	Encodes the data in such a way that if the data being encoded are mostly ASCII text, the encoded form of the data remains largely recognizable by humans.
base64	Encodes data by mapping 6-bit blocks of input to 8-bit blocks of output, all of which are printable ASCII characters.
x-token	A named nonstandard encoding.

A compliant implementation must support the MIME – Version, Content – Type, and Content – Transfer – Encoding fields; the Content – ID and Content-Description fields are optional and may be ignored by the recipient implementation.

Ques 26) What is SNMP? What are the key elements of network management?

Or

List the components of SNMP? **(2019[03])**

Or

Describe SNMP in details with a diagram. (2021[06])

Ans: Simple Network Management Protocol (SNMP)

Simple Network Management Protocol (SNMP) is an application-layer protocol defined by the Internet Architecture Board (IAB) in RFC1157 for exchanging management information between network devices. It is a part of Transmission Control Protocol/Internet Protocol (TCP/IP) protocol suite.

Simple Network Management Protocol (SNMP) is a UDP-based network protocol. It is used mostly in network management systems to monitor network-attached devices for conditions that warrant administrative attention.

Figure 6.22 shows the network management using SNMP. A network management system consists of incremental hardware and software additions implemented among existing network components. The software used in accomplishing the network management tasks resides in the host computers and communications processors (e.g., networks switches, routers).

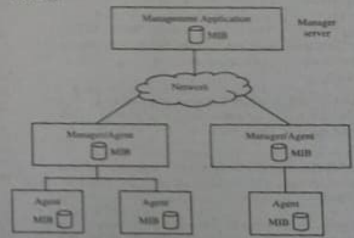


Figure 5.19: Network Management using SNMP

A network management system is designed to view the entire network as a unified architecture, with addresses and labels assigned to each point and the specific attributes of each element and link known to the system. The active elements of the network provide regular feedback of status information to the network control center.