## 1. Explain the services and design issues of Network layer.

Network layer is majorly focused on getting packets from the source to the destination, routing error handling and congestion control.

**Services of Network layer:**

The services which are offered by the network layer are as follows:

### 1. Packetizing

The process of encapsulating the data received from the upper layers of the network (also called payload) in a network layer packet at the source and decapsulating the payload from the network layer packet at the destination is known as packetizing.

The source host adds a header that contains the source and destination address and some other relevant information required by the network layer protocol to the payload received from the upper layer protocol and delivers the packet to the data link layer.

The destination host receives the network layer packet from its data link layer, decapsulates the packet, and delivers the payload to the corresponding upper layer protocol. The routers in the path are not allowed to change either the source or the destination address. The routers in the path are not allowed to decapsulate the packets they receive unless they need to be fragmented.

### 2. Routing

Routing is the process of moving data from one device to another device. These are two other services offered by the network layer. In a network, there are a number of routes available from the source to the destination. The network layer specifies some strategies which find out the best possible route. This process is referred to as routing. There are a number of routing protocols that are used in this process and they should be run to help the routers coordinate with each other and help in establishing communication throughout the network.

### 3. Forwarding

Forwarding is simply defined as the action applied by each router when a packet arrives at one of its interfaces. When a router receives a packet from one of its attached networks, it needs to forward the packet to another attached network (unicast routing) or to some attached networks (in the case of multicast routing). Routers are used on the network for forwarding a packet from the local network to the remote network. So, the process of routing involves packet forwarding from an entry interface out to an exit interface.

### 4. Error Control

Although it can be implemented in the network layer, it is usually not preferred because the data packet in a network layer may be fragmented at each router, which makes error-checking inefficient in the network layer.

### 5. Flow Control

It regulates the amount of data a source can send without overloading the receiver. If the source produces data at a very faster rate than the receiver can consume it, the receiver will be overloaded with data. To control the flow of data, the receiver should send feedback to the sender to inform the latter that it is overloaded with data.

There is a lack of flow control in the design of the network layer. It does not directly provide any flow control. The datagrams are sent by the sender when they are ready, without any attention to the readiness of the receiver.

**6. Congestion Control**

Congestion occurs when the number of datagrams sent by the source is beyond the capacity of the network or routers. This is another issue in the network layer protocol. If congestion continues, sometimes a situation may arrive where the system collapses and no datagrams are delivered. Although congestion control is indirectly implemented in the network layer, still there is a lack of congestion control in the network layer.

**Network layer design issues:**

The network layer comes with some design issues they are described as follows:

**1. Store and Forward packet switching:**

The host sends the packet to the nearest router. This packet is stored there until it has fully arrived once the link is fully processed by verifying the checksum then it is forwarded to the next router till it reaches the destination. This mechanism is called “Store and Forward packet switching.”

**2. Services provided to Transport Layer:**

Through the network/transport layer interface, the network layer transfers it’s services to the transport layer. These services are described below.

But before providing these services to the transfer layer following goals must be kept in mind :

- Offering services must not depend on router technology.
- The transport layer needs to be protected from the type, number and topology of the available router.
- The network addresses for the transport layer should use uniform numbering pattern also at LAN and WAN connections.

Based on the connections there are 2 types of services provided :

- **Connectionless –** The routing and insertion of packets into subnet is done individually. No added setup is required.
- **Connection-Oriented –** Subnet must offer reliable service and all the packets must be transmitted over a single route.

**3. Implementation of Connectionless Service:**

Packet are termed as “datagrams” and corresponding subnet as “datagram subnets”. When the message size that has to be transmitted is 4 times the size of the packet, then the network layer divides into 4 packets and transmits each packet to router via. a few protocol. Each data packet has destination address and is routed independently irrespective of the packets.

**4. Implementation of Connection Oriented service:**

To use a connection-oriented service, first we establishes a connection, use it and then release it. In connection-oriented services, the data packets are delivered to the receiver in the same order in which they have been sent by the sender.

It can be done in either two ways :

- **Circuit Switched Connection –** A dedicated physical path or a circuit is established between the communicating nodes and then data stream is transferred.
- **Virtual Circuit Switched Connection –** The data stream is transferred over a packet switched network, in such a way that it seems to the user that there is a dedicated path from the sender to the receiver. A virtual path is established here. While, other connections may also be using the same path.

### 2. What are the responsibilities of Network layer?

The Network Layer, as defined in the OSI (Open Systems Interconnection) model or the Internet Protocol Suite (TCP/IP), is responsible for several critical functions, often referred to as its "responsibilities." Here are the primary responsibilities of the Network Layer:

**1. Logical Addressing**: One of the fundamental responsibilities of the Network Layer is logical addressing. Devices on a network are assigned logical addresses (such as IP addresses in the case of the Internet). These addresses are used to uniquely identify devices within a network or across networks. Logical addressing allows routers and other networking devices to determine where to forward data packets.

**2. Routing:** Routing is a core function of the Network Layer. It involves determining the optimal path for data packets to travel from the source to the destination. Network routers, which operate at this layer, use routing algorithms to make decisions about how to forward data based on the destination address contained in the packet header. Routing is essential for data to traverse complex networks with multiple intermediate nodes.

**3. Forwarding:** The Network Layer is responsible for forwarding data packets from the source to the destination. This involves examining the destination address in the packet header and making the appropriate forwarding decision based on the routing table.

**4. Fragmentation and Reassembly:** Large data messages or packets may need to be broken down into smaller units, or packets, for transmission over a network. The Network Layer is responsible for fragmenting data when necessary, sending these fragments across the network, and ensuring they are reassembled correctly at the destination.

**5. Error Handling:** The Network Layer also plays a role in error handling. It may detect errors in data packets, such as checksum errors, and decide how to handle these errors, which might involve requesting retransmission of the packet or dropping it.

**6. Congestion Control:** In large and busy networks, congestion can occur when there is more data traffic than the network can handle efficiently. The Network Layer is responsible for implementing congestion control mechanisms to manage and alleviate network congestion, ensuring the smooth flow of data.

**7. Quality of Service (QoS):** The Network Layer may implement Quality of Service mechanisms to prioritize certain types of traffic (e.g., voice or video) over others. QoS ensures that critical applications receive the necessary bandwidth and low latency, even during periods of high network activity.

**8. Tunneling:** In some cases, the Network Layer is responsible for encapsulating packets within another protocol for secure or efficient transmission. This is known as tunneling and is often used in Virtual Private Networks (VPNs) or to traverse networks with different technologies.

**9. Network Address Translation (NAT):** In the context of the Internet, the Network Layer handles NAT, which allows multiple devices within a private network to share a single public IP address for external communication.

Overall, the Network Layer is crucial for the proper functioning of computer networks, as it ensures that data is correctly routed, transmitted, and delivered between devices across the network. It plays a central role in achieving end-to-end communication in both local and global network environments.

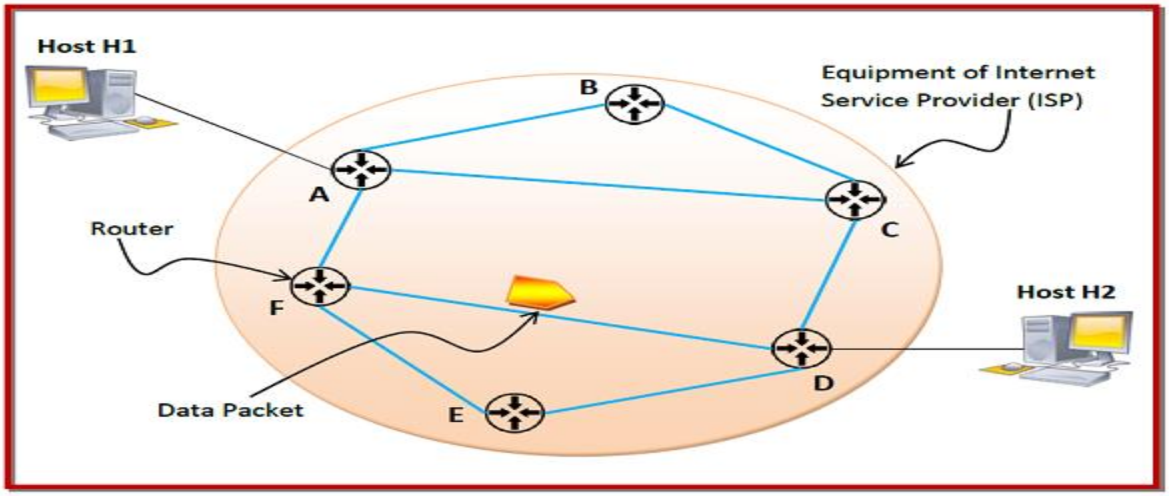
## 3. Explain in brief Store and Forward Packet Switching

In telecommunications, store − and − forward packet switching is a technique where the data packets are stored in each intermediate node, before they are forwarded to the next node. The intermediate node checks whether the packet is error−free before transmitting, thus ensuring integrity of the data packets. In general, the network layer operates in an environment that uses store and forward packet switching.

### Working Principle

The node which has a packet to send, delivers it to the nearest node, i.e. router. The packet is stored in the router until it has fully arrived and its checksum is verified for error detection. Once, this is done, the packet is transmitted to the next router. The same process is continued in each router until the packet reaches its destination.

The following scenario exemplifies the mechanism −



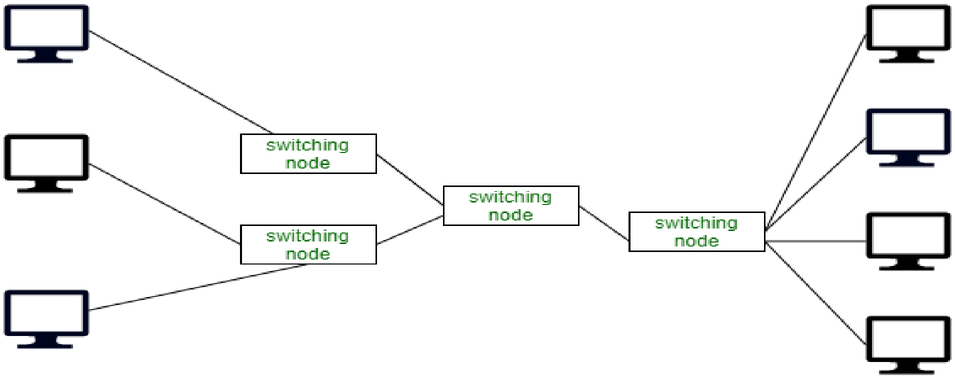
In the above diagram, we can see that the Internet Service Provider (ISP) has six routers (A to F) connected by transmission lines shown in blue lines. There are two hosts, host H1 is connected to router A, while host H2 is connected to router D. Suppose that H1 wants to send a data packet to H2. H1 sends the packet to router A. The packet is stored in router A until it has arrived fully. Router A verifies the checksum using CRC (cyclic redundancy check) code. If there is a CRC error, the packet is discarded, otherwise it is transmitted to the next hop, here router F. The same process is followed by router F which then transmits the packet to router D. Finally router D delivers the packet to host H2.

**Advantages and Disadvantages**

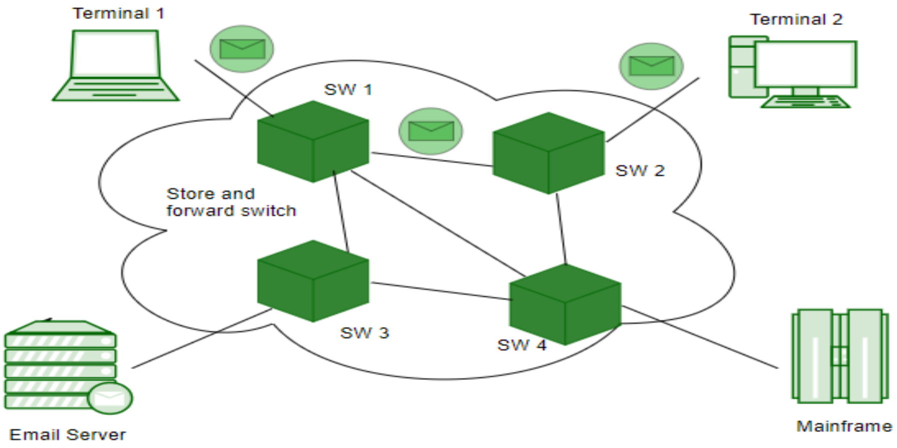
Store − and forward packet switching ensures high quality data packet transmission. Since erroneous packets are discarded at each router, bad packets or invalid packets in the network are mostly eliminated. However, error − free packet transmission is achieved by compromising on the overall speed of transmission. Switch latency is introduced due to waiting for entire packet to arrive as well as computation of CRC. Though the latency at each router may seem small enough, the cumulative latency at all routers make it inappropriate for time − critical online applications.

## 4. Compare circuit switching and message switching

Both **Circuit switching** and **Message switching** are the methods used to connect different devices with each other. The main difference between Circuit switching and Message switching is that Circuit Switching is done by setting a physical path between two systems while Message switching works on the **Store** and **Forward** method.



*Circuit switching*



*Message switching*

In message Switching, data is first stored by one node then forward to another node to transfer the data to another system.

**Difference between Circuit switching and Message switching:**

| S.NO | Circuit Switching | Message Switching |
|---|---|---|
| 1. | It is done by setting a physical path between two systems. | Here, data is first stored by one node then forward to another node to transfer the data to another system. |
| 2. | In circuit switching, data is not stored. | In message Switching, data is first stored, then forwarded to the next node. |
| 3. | Circuit Switching needs a dedicated physical path that's why the messages need not be addressed. | Message switching does not need a dedicated physical path and on Message switching, The messages are addressed independently. |
| 4. | Circuit Switching is **Geographical addressing**. | Message Switching is **Hierarchical addressing**. |
| 5. | Circuit Switching is costlier than message Switching. | The cost of message switching is less than circuit switching. |

| | | |
|---|---|---|
| 6. | Circuit switching routing is manual type routing. | Message Switching routing is not manual type routing, here route is selected during call setup. |
| 7. | Circuit switching reserves the full bandwidth in advance. because of that, there is a lot of wastage of bandwidth | Message Switching does not reserve the entire bandwidth in advance. and that’s why bandwidth is used to its maximum extent. |
| 8. | In-circuit switching, the charge depends on time and distance. | In message switching, the charge is based on the number of bytes and distance. |
| 9. | Congestion occurs per minute in circuit switching. | In message switching, no congestion or very little congestion occurs. |
| 10. | Circuit switching uses Analog and digital media on a variety of platforms. | Whereas Message switching uses digital media on a variety of platforms. |
| 11. | In-circuit switching there is no propagation delay. | While In Message switching there is a propagation delay. |
| 12. | The transmission capacity of circuit switching is very low. | while the transmission capacity of message switching is high. |
| 13. | In a circuit switching, Messages need not be addressed as there is one dedicated path. | In message switching, Messages are addressed as independent routes are established. |
| 14. | Ex. Real time transfer of voice signals. | Ex. Transmission of telegram. |

### 5. With a neat sketch, explain the following in detail Circuit Switching Technique and Packet Switching Technique.

### Circuit Switching

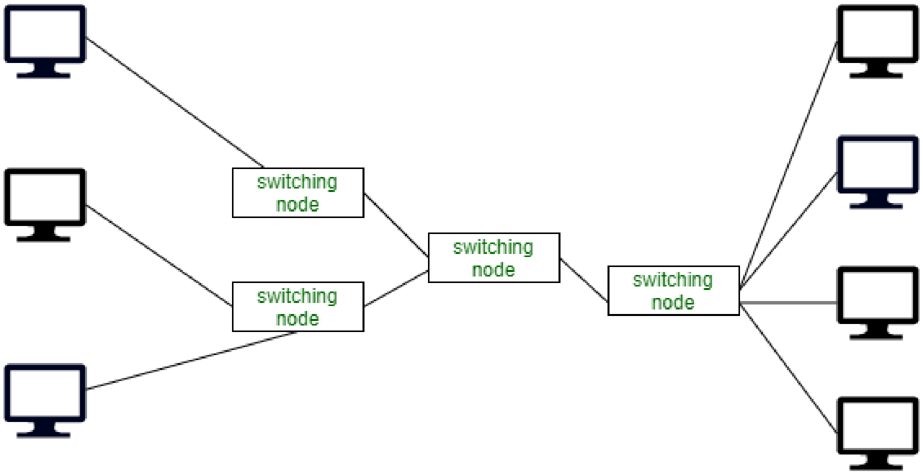
**It** is a connection-oriented service. It provides a dedicated path from the sender to the receiver. In-circuit switching, a connection setup is required to send and receive data. It has very little chance of data loss and error due to the dedicated circuit, but a lot of bandwidth is wasted because the same path cannot be used by other senders during a congestion. Circuit switching is completely transparent; the sender and receiver can use any bit rate format or framing method.

### Advantages of Circuit Switching

- It uses a fixed bandwidth.
- A dedicated communication channel increases the quality of communication.
- Data is transmitted with a fixed data rate.
- No waiting time at switches.
- Suitable for long continuous communication.

**Disadvantages of circuit switching**

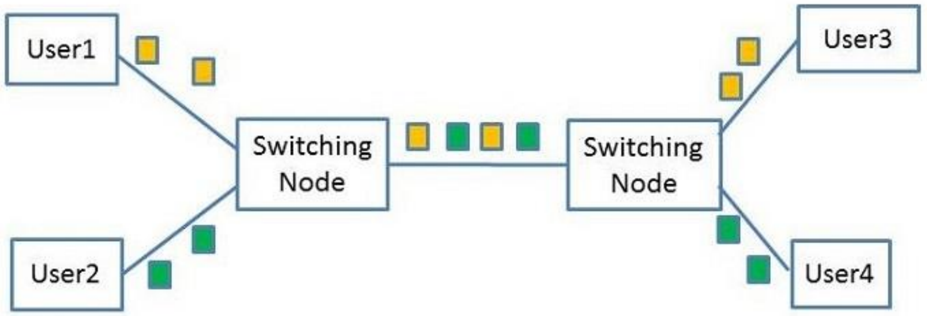
- A dedicated connection makes it impossible to transmit other data even if the channel is free.
- Resources are not utilized fully.
- The time required to establish the physical link between the two stations is too long.
- A dedicated path has to be established for each connection.
- Circuit switching is more expensive.



*Circuit Switching*

**Packet Switching**

It is a connectionless service. It does not require any dedicated path between the sender and receiver. It places an upper limit on block size. In packet switching bandwidth is freely utilized as unrelated sources can be used in any path. It has more chance of data loss and error; the packets may arrive in the wrong order.



Packet Switching

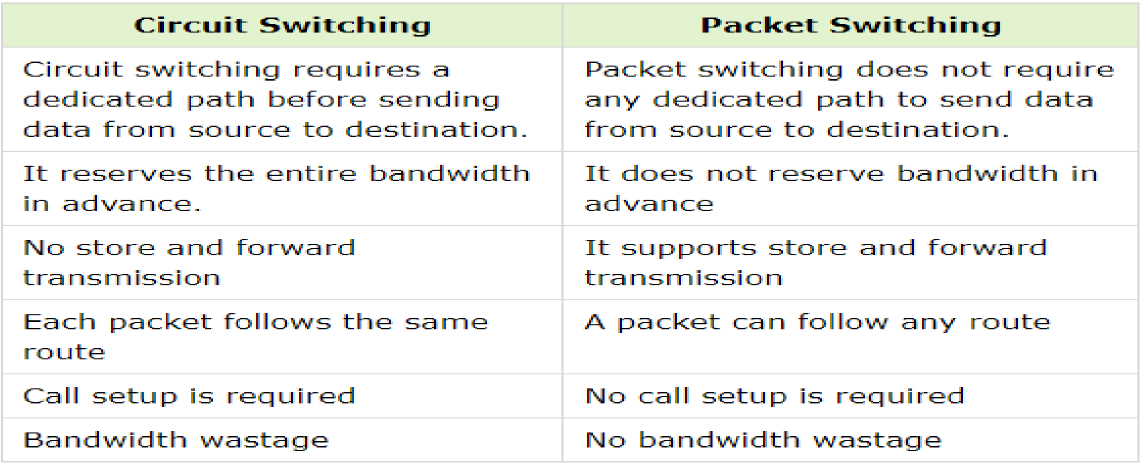
## Advantages of Packet switching

- It reduces access delay.
- Costs are minimized to great extent. Hence packet switching is a very cost-effective technique.
- Packets are rerouted in case of any problems. This ensures reliable communication.
- It is more efficient for data transmission because no need to establish the path.
- Several users can share the same channel simultaneously. Therefore packet switching makes use of available bandwidth efficiently.

## Disadvantages of Packet switching

- In packet switching, the network cannot be used in applications requiring very little delay and higher quality of service.
- Protocols used in the packet switching are complex.
- If the network becomes overloaded, packets are delayed or discarded, or dropped. This leads to the retransmission of lost packets by the sender.

## Difference between circuit switching and packet switching



| Circuit Switching | Packet Switching |
|---|---|
| Circuit switching requires a dedicated path before sending data from source to destination. | Packet switching does not require any dedicated path to send data from source to destination. |
| It reserves the entire bandwidth in advance. | It does not reserve bandwidth in advance |
| No store and forward transmission | It supports store and forward transmission |
| Each packet follows the same route | A packet can follow any route |
| Call setup is required | No call setup is required |
| Bandwidth wastage | No bandwidth wastage |

## 6. Discuss in detail Virtual circuit vs Datagram sockets.

Virtual circuit and datagram sockets represent two different communication paradigms in networking. Each has its own set of characteristics and use cases. Let's discuss them in detail:

### Virtual Circuit:

**1. Connection-Oriented:** Virtual circuit is a connection-oriented communication model. Before data transfer begins, a dedicated path or circuit is established between communicating parties. This path is maintained for the duration of the communication.

**2. Setup Phase:** The process begins with a setup phase where a connection is established. During this phase, information about the path is exchanged between the sender and receiver. This setup is often analogous to making a reservation for a dedicated line.

**3. Reliability:** Virtual circuits are highly reliable. Once the connection is established, data is sent over this pre-determined path. This approach is similar to circuit-switched telephone networks and ensures that data is delivered in the correct order and without loss.

**4. Orderly Delivery:** Data is delivered in an orderly and predictable manner, making it suitable for applications that require strict sequencing, such as voice and video communication.

**5. Resource Reservation:** Resources along the path (e.g., bandwidth) are reserved for the duration of the communication, guaranteeing consistent quality of service.

**6. Examples:** Frame Relay and Asynchronous Transfer Mode (ATM) are examples of protocols that use virtual circuits.

### Datagram Sockets:

**1. Connectionless:** Datagram sockets are connectionless, which means there is no setup phase or dedicated path before data transfer. Each packet (datagram) is sent independently and may take different routes to reach the destination.

**2. No Path Reservation:** Unlike virtual circuits, there is no reservation of resources or dedicated path. Datagram packets can share the network with other packets, and resources are allocated on a per-packet basis.

**3. Flexibility:** Datagram sockets are highly flexible. They are well-suited for applications with varying data sizes and sporadic communication. Each packet is independent, and the network can adapt to changes in traffic patterns.

**4. Reliability Not Guaranteed:** Datagram sockets do not guarantee reliable data delivery. Packets may arrive out of order or be lost. While this might not be suitable for some applications, it's acceptable for others, like real-time video streaming.

**5. Examples:** The Internet Protocol (IP) is the most common example of a protocol that uses datagram sockets. UDP (User Datagram Protocol) operates at the transport layer and is connectionless, making it suitable for datagram-style communication.

### Use Cases:

**Virtual Circuit:** Virtual circuits are often used in applications where reliability and strict sequencing are critical, such as voice over IP (VoIP) calls or video conferencing, as well as in older technologies like ATM and Frame Relay networks.

**Datagram Sockets:** Datagram sockets are suitable for applications where flexibility and lower overhead are more important than reliability. Examples include real-time online gaming, video streaming, DNS (Domain Name System) queries, and many Internet-based applications.

### 7. Differences between static routing and dynamic routing with examples (Adaptive and non-Adaptive routing).

| S.NO | Static Routing | Dynamic Routing |
|---|---|---|
| 1. | In static routing routes are user-defined. | In dynamic routing, routes are updated according to the topology. |
| 2. | Static routing does not use complex routing algorithms. | Dynamic routing uses complex routing algorithms. |
| 3. | Static routing provides high or more security. | Dynamic routing provides less security. |
| 4. | Static routing is manual. | Dynamic routing is automated. |
| 5. | Static routing is implemented in small networks. | Dynamic routing is implemented in large networks. |
| 6. | In static routing, additional resources are not required. | In dynamic routing, additional resources are required. |
| 7. | In static routing, failure of the link disrupts the rerouting. | In dynamic routing, failure of the link does not interrupt the rerouting. |
| 8. | Less Bandwidth is required in Static Routing. | More Bandwidth is required in Dynamic Routing. |
| 9. | Static Routing is difficult to configure. | Dynamic Routing is easy to configure. |

|  |  |  |
|---|---|---|
| 10. | Another name for static routing is non-adaptive routing. | Another name for dynamic routing is adaptive routing. |

**8. Compare Adaptive and Non-adaptive routing algorithms.**

**1. Adaptive Routing algorithm:**
Adaptive routing algorithm is also called a dynamic routing algorithm. In this algorithm, the routing decisions are made based on network traffic and topology. The parameters which are used in adaptive routing algorithms are distance, hop, estimated transit time and count.

The adaptive routing algorithm is of three types –

- Centralized algorithm
- Isolation algorithm
- Distributed algorithm

**2. Non-Adaptive Routing algorithm:**
Non-adaptive routing algorithm is also called a static routing algorithm. In a non-adaptive routing algorithm, the routing decisions are not made based on network traffic and topology. This algorithm is used by static routing. Non-adaptive routing algorithms are simple as compared to Adaptive routing algorithms in terms of complexity.

The non-adaptive routing algorithm is of two types –

- Flooding
- Random walks

**Difference between Adaptive and Non-Adaptive Routing algorithms:**

| **S. No.** | **Adaptive Routing algorithm** | **Non-Adaptive Routing algorithm** |
|---|---|---|
| 1. | An adaptive algorithm involves routers for exchanging and updating router table data. | A non-adaptive algorithm involves a network administrator for the manual entry of the routing paths into the router. |
| 2. | This algorithm creates a routing table based on network conditions. | Whereas this algorithm creates a static table in order to determine when to send packets and which node. |

| | | |
|---|---|---|
| 3. | This algorithm is used by dynamic routing. | Whereas this algorithm is used by static routing. |
| 4. | In adaptive routing algorithm, the routing decisions are made based on network traffic and topology. | Whereas in a non-adaptive routing algorithm, the routing decisions are not made based on network traffic and topology. |
| 5. | Adaptive routing algorithms are more complex as compared to non-adaptive routing algorithms in terms of complexity. | While non-adaptive routing algorithms are simple in terms of complexity. |
| 6. | In adaptive routing algorithm, the routing decisions are not static tables. | While in non-adaptive routing algorithm, the routing decisions are static tables. |
| 7. | Adaptive routing algorithm is categorized into distributed, centralized and isolation algorithm. | Whereas non-adaptive routing algorithm is categorized into random walks and flooding. |
| 8. | Adaptive routing algorithm is more used as compared to non-adaptive. | Whereas non-adaptive routing algorithm is comparatively less used. |
| 9. | The dynamic protocols are employed to update the routing table and determine the best route between the source and destination computers. | The manual setup is performed for establishing an optimal path between the source and destination computers. |
| 10. | It is mostly used for-<br>• Open, Complex network topologies | It is mostly used for-<br>• Simple, Closed network topologies |
| 11. | Purposes-<br>• Enhancement in network performance<br>• Prevents packet delivery failure<br>• Aid in controlling congestion | Purposes-<br>• It enables fine-grained control over packet paths.<br>• Suited for reliable networks with stable loads |

**9. Compare connection oriented and connection less services provided by the network layer.**

| S.NO | Connection-oriented Service | Connection-less Service |
|---|---|---|
| 1. | Connection-oriented service is related to the telephone system. | Connection-less service is related to the postal system. |
| 2. | Connection-oriented service is preferred by long and steady communication. | Connection-less Service is preferred by bursty communication. |
| 3. | Connection-oriented Service is necessary. | Connection-less Service is not compulsory. |
| 4. | Connection-oriented Service is feasible. | Connection-less Service is not feasible. |
| 5. | In connection-oriented Service, Congestion is not possible. | In connection-less Service, Congestion is possible. |
| 6. | Connection-oriented Service gives the guarantee of reliability. | Connection-less Service does not give a guarantee of reliability. |
| 7. | In connection-oriented Service, Packets follow the same route. | In connection-less Service, Packets do not follow the same route. |
| 8. | Connection-oriented services require a bandwidth of a high range. | Connection-less Service requires a bandwidth of low range. |
| 9. | Ex: TCP (Transmission Control Protocol) | Ex: UDP (User Datagram Protocol) |
| 10. | Connection-oriented requires authentication. | Connection-less Service does not require authentication. |

**10. How is the Connection-Oriented Services implemented? Explain**

**Connection-Oriented Service** is basically a technique that is typically used to transport and send data at session layer. The data streams or packets are transferred or delivered to receiver in a similar order in which they have seen transferred by sender. It is actually a data transfer method among two devices or computers in a different network, that is designed and developed after

telephone system. Whenever a network implements this service, it sends or transfers data or message from sender or source to receiver or destination in correct order and manner.
This connection service is generally provided by protocols of both network layer (signifies different path for various data packets that belongs to same message) as well as transport layer (use to exhibits independence among packets rather than different paths that various packets belong to same message will follow).

**Operations:**
There is a sequence of operations that are needed to b followed by users. These operations are given below:

1. **Establishing Connection –**
   It generally requires a session connection to be established just before any data is transported or sent with a direct physical connection among sessions.
2. **Transferring Data or Message –**
   When this session connection is established, then we transfer or send message or data.
3. **Releasing the Connection –**
   After sending or transferring data, we release connection.

**Different Ways:**
There are two ways in which connection-oriented services can be done. These ways are given below:

1. **Circuit-Switched Connection –**
   Circuit-switching networks or connections are generally known as connection-oriented networks. In this connection, a dedicated route is being established among sender and receiver, and whole data or message is sent through it. A dedicated physical route or a path or a circuit is established among all communication nodes, and after that, data stream or message is sent or transferred.
2. **Virtual Circuit-Switched Connection –**
   Virtual Circuit-Switched Connection or Virtual Circuit Switching is also known as Connection-Oriented Switching. In this connection, a pre-planned route or path is established before data or messages are transferred or sent. The message Is transferred over this network is such a way that it seems to user that there is a dedicated route or path from source or sender to destination or receiver.

**Types of Connection-Oriented Service:**

| **Service** | **Example** |
|---|---|
| Reliable Message Stream | Sequence of pages, etc. |
| Reliable Byte Stream | Song Download, etc. |
| Unreliable Connection | VoIP (Voice Over Internet Protocol) |

**11) What is Routing Algorithm? Briefly discuss Adaptive Routing Algorithms and Non – Adaptive Routing Algorithms.**

A routing algorithm is a set of rules and protocols used in computer networks to determine the path that data should take from its source to its destination. Routing algorithms play a crucial role in ensuring efficient and reliable data transmission in network communication. They are responsible for making decisions about how to forward data packets through the network.

There are two primary categories of routing algorithms: adaptive routing algorithms and non-adaptive routing algorithms.

1. Adaptive Routing Algorithms:
    - Adaptive routing algorithms are dynamic and can adjust their routing decisions based on the current network conditions and traffic loads. They continually monitor the state of the network and make routing decisions on the fly to optimize data transmission.
    - These algorithms are more flexible and responsive to changes in the network, which can help in load balancing and avoiding congested paths.
    - Examples of adaptive routing algorithms include:
    - a. Distance-Vector Routing: In this approach, routers exchange information about the distances to various destinations, and the routing tables are periodically updated based on this information. Routing decisions consider the shortest path.
    - b. Link-State Routing: Routers exchange information about the state and cost of each link in the network. This information is used to build a comprehensive view of the network topology, and routing decisions are made based on this topology.
2. Non-Adaptive Routing Algorithms:
    - Non-adaptive routing algorithms, also known as static routing algorithms, make routing decisions based on fixed, predetermined rules and tables. These rules do not change in response to network conditions or traffic load.
    - Non-adaptive algorithms are typically simpler and easier to implement, but they may not always result in the most efficient paths, especially in dynamic network environments.
    - Examples of non-adaptive routing algorithms include:

- a. Shortest Path Routing: This is a type of non-adaptive routing where the shortest path is predetermined and remains constant. It's suitable for networks with stable topologies.
- b. Fixed Routing: In fixed routing, specific routes are manually configured and remain unchanged unless manually updated. It's commonly used in small, predictable networks.

In summary, routing algorithms are a crucial part of network communication, helping to determine how data packets travel from source to destination. Adaptive routing algorithms adjust their decisions based on real-time network conditions, while non-adaptive routing algorithms rely on predetermined routes. The choice of routing algorithm depends on the specific network requirements and the desired balance between adaptability and simplicity.

### 12. Compare Distance Vector Routing and Link State Routing.

| S.No | **Distance Vector Routing** | **Link State Routing** |
|---|---|---|
| 1. | Bandwidth required is less due to local sharing, small packets and no flooding. | Bandwidth required is more due to flooding and sending of large link state packets. |
| 2. | Based on local knowledge, since it updates table based on information from neighbours. | Based on global knowledge, it have knowledge about entire network. |
| 3. | Make use of Bellman Ford Algorithm. | Make use of Dijakstra's algorithm. |
| 4. | Traffic is less. | Traffic is more. |
| 5. | Converges slowly i.e, good news spread fast and bad news spread slowly. | Converges faster. |
| 6. | Count of infinity problem. | No count of infinity problem. |
| 7. | Practical implementation is RIP and IGRP. | Practical implementation is OSPF and ISIS. |
| 8. | Persistent looping problem i.e, loop will be there forever. | No persistent loops, only transient loops. |

## 13) With an example explain the Dijkstra’s (Shortest path) routing algorithms used in computer networks

Dijkstra's algorithm is a widely used routing algorithm in computer networks for finding the shortest path from a source node to all other nodes in a weighted graph. This algorithm is particularly useful in situations where you want to determine the most efficient path to reach a destination node while considering the cost associated with traversing each link or edge in the network. Let's explain Dijkstra's algorithm with an example:

Suppose you have a computer network with the following topology:

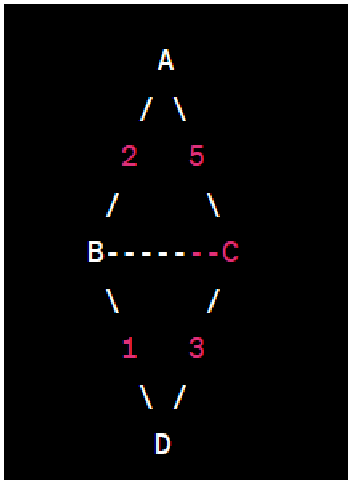
In this network, nodes A, B, C, and D are connected by links with corresponding weights (the numbers on the edges). We want to find the shortest path from node A to all other nodes in the network using Dijkstra's algorithm.

Here's how Dijkstra's algorithm works:

1. Initialize a list of distances from the source node (A) to all other nodes. Initially, set the distance to A as 0 and all other distances to infinity. Also, maintain a list of unvisited nodes.

Distance: [A: 0, B: ∞, C: ∞, D: ∞]

Unvisited: [A, B, C, D]



```
      A
     / \
    2   5
   /     \
  B-------C
   \     /
    1   3
     \ /
      D
```

2. Select the node with the smallest distance (in this case, A with a distance of 0) and mark it as visited.

   Visited: [A]
   Unvisited: [B, C, D]
3. Update the distances of adjacent nodes that are unvisited. Calculate the sum of the distance from the source to the currently selected node and the weight of the edge to the adjacent node. If this distance is smaller than the current known distance to the adjacent node, update the distance.

   Distance: [A: 0, B: 2, C: 5, D: ∞]

4. Repeat steps 2 and 3 until all nodes are visited.
    1. Select node B with a distance of 2.
    2. Update distances: A → B (2)
    3. Distance: [A: 0, B: 2, C: 3, D: ∞]
    4. Select node C with a distance of 3.
    5. Update distances: A → C (3)
    6. Distance: [A: 0, B: 2, C: 3, D: 6]
    7. Select node B with a distance of 2.
    8. Update distances: C → D (6)
    9. Distance: [A: 0, B: 2, C: 3, D: 9]
5. When all nodes are visited or marked as unreachable, the algorithm is complete. The final distances represent the shortest paths from node A to all other nodes in the network.

Shortest Paths:

A to A: 0

A to B: 2

A to C: 3

A to D: 9

In this example, Dijkstra's algorithm found the shortest paths from node A to all other nodes in the network. The shortest path from A to D is A → B → C → D with a total cost of 9.

**14) Explain the Hierarchical Routing algorithm and discuss its advantages and limitations**

Hierarchical routing is a network routing algorithm that organizes large networks into hierarchical structures to improve scalability, reduce routing overhead, and simplify network management. It is commonly used in complex networks like the internet and large corporate networks. Here's an explanation of hierarchical routing, along with its advantages and limitations:
Hierarchical Routing Algorithm:
In hierarchical routing, the network is divided into multiple levels or hierarchies. Each level represents a smaller administrative or geographical region. At each level, there is a designated routing entity (e.g., a router or a routing domain) responsible for routing within that region. These regional routers are often called "core routers."

- Advantages:

1. Scalability: One of the primary advantages of hierarchical routing is improved scalability. In large networks, it's impractical for every router to maintain a complete routing table

with information about all other routers in the network. Hierarchical routing divides the network into smaller manageable regions, reducing the size of routing tables at individual routers.

2. Reduced Routing Overhead: With hierarchical routing, routers within the same hierarchy only need to maintain information about routes within their region. This significantly reduces the amount of routing information that needs to be exchanged and processed, which, in turn, decreases routing overhead.
3. Efficient Route Aggregation: Hierarchical routing allows for efficient aggregation of routes. Instead of advertising individual routes for every device, a hierarchical router can advertise a summary route for an entire region. This reduces the number of routing table entries and simplifies the routing process.
4. Enhanced Network Management: By dividing the network into hierarchies, network administrators can more effectively manage and control different parts of the network. This separation of responsibilities can lead to improved network organization and maintenance.

- **Limitations:**

1. Hierarchical Structure Complexity: Setting up and maintaining the hierarchical structure itself can be complex. Determining the appropriate hierarchy levels, addressing schemes, and the assignment of regional routers requires careful planning.
2. Hierarchical Design Updates: When network topology or traffic patterns change, adapting the hierarchical design can be challenging. Reconfiguring hierarchies or adding new regions may require significant effort and network downtime.
3. Single Point of Failure: Core routers or regional routers at the top of the hierarchy can become single points of failure. If a core router fails, it can disrupt communication across multiple regions.
4. Latency: Routing through multiple hierarchy levels can introduce additional latency in the network. This is a trade-off for the benefits of scalability and reduced routing overhead.
5. Limited Adaptability: Hierarchical routing may not be suitable for highly dynamic networks, where traffic patterns and network topology frequently change. It is better suited for relatively stable network environments.

In summary, hierarchical routing is a useful approach for managing large and complex networks efficiently. It addresses the challenges of scalability and routing overhead. However, its design and management complexities, potential single points of failure, and latency considerations need to be carefully weighed against the advantages to determine its suitability for a given network.

15) **Explain distance vector routing algorithm (DVMRP), Describe the problem and solutions associated with distance vector routing.**

Distance Vector Routing Protocol (DVRP), not to be confused with DVMRP (Distance Vector Multicast Routing Protocol), is a class of routing algorithms used in computer networks. Distance vector routing algorithms determine the best path to a destination based on distance or "cost" metrics. One of the most well-known distance vector routing protocols is the Routing Information Protocol (RIP). Distance vector routing algorithms, discuss common problems associated with them, and their solutions:

**Distance Vector Routing Algorithm:**

Distance vector routing algorithms operate by periodically exchanging routing tables with neighbouring routers to calculate the best path to reach various destinations. Each router maintains a table that contains distance metrics to different destinations and the next-hop router to reach them. The routing table entries are updated based on received updates from neighbouring routers.

Common Problems Associated with Distance Vector Routing:

1. **Count to Infinity:** Count to Infinity is a classic problem in distance vector routing algorithms. It occurs when routers in the network believe they have a route to a destination even after the actual route has gone down. This can result in routing loops and incorrect path selection.
    - Problem: Suppose Router A thinks it has a route to a destination through Router B. If Router B's link to the destination goes down, it doesn't immediately inform Router A. Router A still believes it has a path through Router B, so it continues to send packets to the unreachable destination, causing a loop.
    - Solution: Techniques like split horizon and route poisoning are used to mitigate the Count to Infinity problem. Split horizon prevents a router from advertising a route back to the router it learned it from, and route poisoning involves advertising unreachable routes with infinite metric values to inform other routers about link failures.
2. **Slow Convergence**: Distance vector algorithms can be slow to converge when the network topology changes. Convergence refers to the process of routers adjusting their routing tables to reflect changes in the network. Slow convergence can result in packets being dropped or taking suboptimal paths.
    - Problem: If a link in the network goes down, routers may take multiple iterations to update their routing tables and propagate the new information to other routers.
    - Solution: To improve convergence time, various techniques have been developed, such as triggered updates (immediately sending route updates when changes are detected) and route aggregation (reducing the number of route updates sent).
3. **Routing Loops:** Routing loops can occur when routers continuously update each other with conflicting information about a destination, causing packets to circulate indefinitely.
    - Problem: If Router A thinks Router B is the best path to a destination, and Router B thinks Router A is the best path, they may keep updating each other with their routes, causing a loop.

- Solution: Techniques like split horizon and route poisoning, as mentioned earlier, help mitigate routing loops. Additionally, using a maximum hop count or time-to-live (TTL) for packets can prevent them from circulating infinitely.

In summary, distance vector routing algorithms have their advantages, including simplicity and low computational overhead, but they also come with common problems like Count to Infinity, slow convergence, and routing loops. These issues can be mitigated through various mechanisms such as split horizon, route poisoning, triggered updates, and improved algorithms like the Enhanced Interior Gateway Routing Protocol (EIGRP) and the Border Gateway Protocol (BGP).

**16) Explain Link state routing algorithm (or ) Illustrate Link State Routing.**

Certainly, let's illustrate how a Link State Routing Algorithm works using a simplified example. In this scenario, we have a small network with four routers: A, B, C, and D.

1. **Link-State Advertisement (LSA) Generation:**
    - Each router initially generates its Link-State Advertisement (LSA) containing information about its directly connected links. The information includes the link's state (up or down) and the cost associated with the link. These LSAs are exchanged among routers.
    - Router A generates an LSA with information about its direct links to B and C.
    - Router B generates an LSA with information about its links to A and C.
    - Router C generates an LSA with information about its links to A, B, and D.
    - Router D generates an LSA with information about its link to C.
2. **Link-State Database (LSDB) Creation:**
    - Each router maintains a Link-State Database (LSDB) to store the LSAs received from its neighbors. The LSDB stores a comprehensive view of the network's topology.
    - The LSDB entries for each router are updated as LSAs are exchanged.
3. **Shortest Path Calculation:**
    - Using the information in the LSDB, routers perform a shortest path calculation to determine the best paths to all destinations in the network. They typically use Dijkstra's algorithm.
    - The cost associated with each link is considered in the calculation.
4. **Routing Table Construction:**
    - After calculating the shortest paths, routers create their routing tables. These tables list the best path to reach each destination within the network.
    - The routing table entries include the next hop router and the cumulative cost to reach that destination.
5. **Periodic LSA Updates:**

    - The network continuously monitors the state of its links. If a link's state changes (e.g., a link goes down), the affected router generates a new LSA and floods it through the network.
    - When a router receives a new LSA or detects changes in the LSDB, it triggers a recalculation of the shortest paths and updates its routing table.
6. **Routing and Data Transmission:**
    - Once the routing tables are constructed, routers use these tables to determine the best path for forwarding data packets. The data packets follow the path defined in the routing table entries.
    - If a link or router fails, the network can quickly adapt by updating LSAs and rerouting packets along alternative paths.

In our simplified example, the Link State Routing Algorithm facilitates the exchange of LSAs among routers, which leads to an accurate view of the network's topology. This information is then used to calculate the shortest paths, allowing routers to make optimal routing decisions. As the network topology changes, routers respond by generating new LSAs, recalculating paths, and updating their routing tables, ensuring efficient data transmission.

17) **Explain flooding with examples.**

Flooding is a network communication technique used to send data packets from a source to all possible destinations without any prior knowledge of the network's topology. In flooding, the source node broadcasts the data packet to all its neighbouring nodes, which, in turn, rebroadcast it to their neighbours, and this process continues until the packet reaches the desired destination. Flooding is a straightforward and robust method but can lead to some inefficiencies, such as packet duplication and excessive network traffic. Here's an explanation of flooding with an example:

**Flooding Process:**

1. **Source Node:** The source node, which wants to send a data packet to a particular destination, initiates the process. It doesn't have any information about the network's topology or the specific path to the destination.
2. **Broadcast:** The source node broadcasts the data packet to all its neighbouring nodes. This means it sends a copy of the packet to every node it is directly connected to.
3. **Neighbouring Nodes:** Each neighbouring node, upon receiving the packet, rebroadcasts it to all of its neighbours. This results in the packet being sent to all nodes connected to these neighbouring nodes, including the source node. However, nodes typically maintain a record of the packets they have already received to avoid infinite loops.
4. **Propagation:** The process continues as each node that receives the packet broadcasts it to its neighbours, and the packet propagates throughout the network.
5. **Destination Node:** If the destination node is within the network, it will eventually receive the packet. When it does, it recognizes the packet as intended for it and processes

it. The destination node may also generate an acknowledgment back to the source, indicating that the data packet was received successfully.

**Example: Broadcast Storm in Ethernet LAN:**

One real-world example of flooding is the "broadcast storm" issue that can occur in Ethernet LANs. In Ethernet, broadcast packets are sent to all devices on a local network segment. If a device on the network generates a broadcast packet, it is flooded to all other devices. If a network has a loop, for example, due to a misconfigured switch or hub, a single broadcast packet can lead to a broadcast storm:

- **Scenario:** Let's say there is a misconfigured switch that creates a loop in the LAN. When a device sends a broadcast packet (e.g., ARP request or DHCP request), the switch forwards it to all other devices. However, due to the loop, the packet arrives back at the switch and is once again broadcast to all devices, creating a loop of endless broadcasts.
- **Impact:** This continuous flood of broadcast packets can overwhelm the LAN, consuming network bandwidth and causing a significant increase in network traffic. It can slow down the network's performance and potentially render it unusable.

To mitigate the broadcast storm issue, network administrators use techniques like Spanning Tree Protocol (STP) to prevent loops in Ethernet networks and employ network segmentation to isolate broadcast domains, reducing the scope of broadcast traffic.

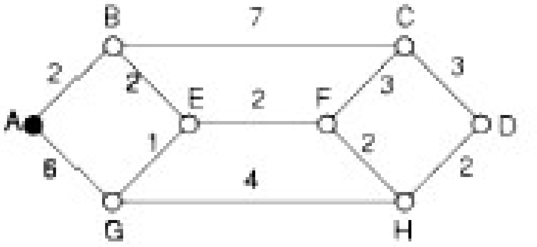
**18) Briefly explain Hierarchical, Broadcast, multicast routing techniques.**

1. **Hierarchical Routing:**
    - Description: Hierarchical routing organizes large networks into hierarchical structures, with routers at different levels responsible for specific regions or domains. Each level of hierarchy manages routing within its domain, reducing the complexity of individual routers.
    - Key Characteristics: Improved scalability, reduced routing overhead, simplified network management.
    - Use Cases: Large corporate networks, the Internet, and other complex network environments.
2. **Broadcast Routing:**
    - Description: Broadcast routing involves sending data packets to all devices in the network. Broadcasts are messages that are intended for every device on the local network segment.
    - Key Characteristics: Simple and easy to implement, suitable for small networks, generates high network traffic.
    - Use Cases: Simple LANs, Address Resolution Protocol (ARP) for mapping IP addresses to MAC addresses.
3. **Multicast Routing:**

- Description: Multicast routing allows data to be sent to a selected group of devices instead of all devices in the network. It efficiently delivers data to recipients who have expressed interest in receiving it.
- Key Characteristics: Efficient use of network resources, reduces network traffic, suitable for one-to-many or many-to-many communication.
- Use Cases: Video streaming, online gaming, content distribution networks (CDNs), and applications where content is shared with multiple recipients.

These routing techniques serve different purposes and are chosen based on the specific requirements and characteristics of the network and the type of communication needed.

19) **Discuss the shortest path routing algorithm with an example**



**20) Distinguish between transparent fragmentation and non-transparent fragmentation**

The key distinctions between transparent fragmentation and non-transparent fragmentation:

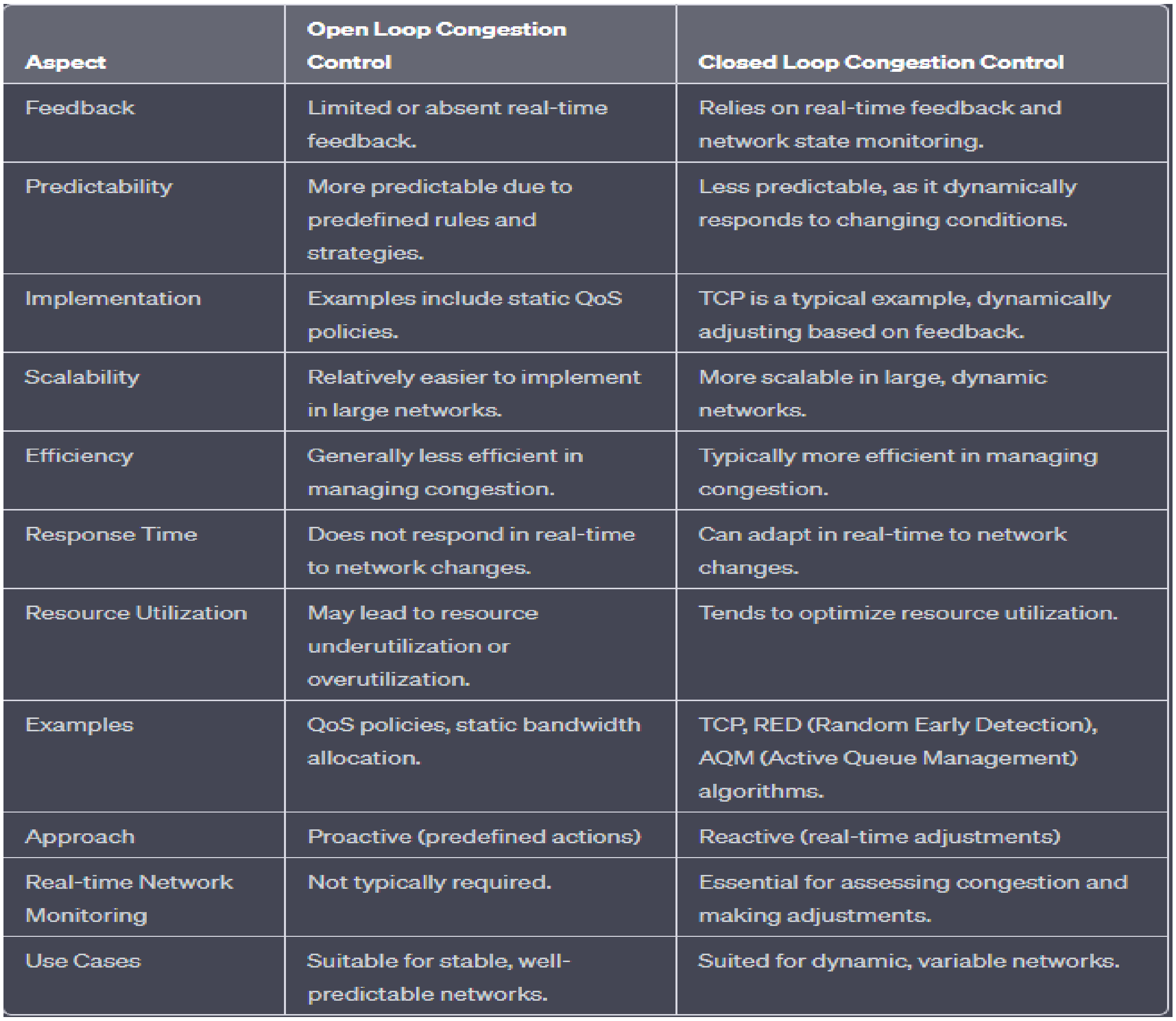
| Aspect | Transparent Fragmentation | Non-Transparent Fragmentation |
|---|---|---|
| **Definition** | Involves breaking data into smaller pieces (fragments) without the sender or receiver being aware that fragmentation has occurred. | Involves dividing data into smaller fragments where both the sender and receiver are aware of the fragmentation process. |
| **Fragmentation Responsibility** | Performed by network devices (routers) along the data transmission path without the knowledge or consent of the sender or receiver. | Typically initiated by the sender, who explicitly decides how to divide the data into fragments. |
| **Fragment Reassembly** | Fragments are reassembled at the receiver without the receiver being aware that the original data was fragmented. | The receiver actively participates in the reassembly process, knowing that fragmentation has occurred. |

| | | |
|---|---|---|
| **Examples** | IP fragmentation in computer networking, where routers can fragment large IP packets into smaller fragments for transmission, and the receiver reassembles them. | HTTP Chunked Transfer-Encoding in web communications, where the sender explicitly divides data into chunks, and the receiver combines them. |
| **Complexity** | Typically simpler for the sender as it doesn't need to manage fragmentation, but potentially more complex for routers and the receiver due to reassembly. | More complex for the sender as it has to manage fragmentation explicitly, but potentially simpler for the receiver as it knows the fragmentation details. |
| **Header Overhead** | Typically less header overhead as only routers need to add fragmentation-related headers. | May involve additional headers in data chunks, such as the "chunk size" indicator in Chunked Transfer-Encoding. |
| **Efficiency and Control** | Less control for the sender over fragmentation and reassembly, which might lead to suboptimal transmission. | More control for the sender over how data is divided and reassembled, allowing for optimization. |
| **Scalability** | More suitable for scenarios where routers need to handle fragmentation on behalf of senders and receivers, but less efficient for sender control. | Suitable for scenarios where the sender and receiver need control and awareness of fragmentation, especially in application-level protocols. |

In summary, transparent fragmentation occurs without the sender or receiver's explicit involvement or awareness, while non-transparent fragmentation involves sender-initiated and receiver-aware fragmentation. The choice between the two depends on factors like network requirements, sender control, and the specific application or protocol in use.

**21. Differentiate the open loop congestion control and closed loop congestion control in detail. [7M] [Dec/Jan202223] Analyse CO2.**

Ans:



| Aspect | Open Loop Congestion Control | Closed Loop Congestion Control |
|---|---|---|
| Feedback | Limited or absent real-time feedback. | Relies on real-time feedback and network state monitoring. |
| Predictability | More predictable due to predefined rules and strategies. | Less predictable, as it dynamically responds to changing conditions. |
| Implementation | Examples include static QoS policies. | TCP is a typical example, dynamically adjusting based on feedback. |
| Scalability | Relatively easier to implement in large networks. | More scalable in large, dynamic networks. |
| Efficiency | Generally less efficient in managing congestion. | Typically more efficient in managing congestion. |
| Response Time | Does not respond in real-time to network changes. | Can adapt in real-time to network changes. |
| Resource Utilization | May lead to resource underutilization or overutilization. | Tends to optimize resource utilization. |
| Examples | QoS policies, static bandwidth allocation. | TCP, RED (Random Early Detection), AQM (Active Queue Management) algorithms. |
| Approach | Proactive (predefined actions) | Reactive (real-time adjustments) |
| Real-time Network Monitoring | Not typically required. | Essential for assessing congestion and making adjustments. |
| Use Cases | Suitable for stable, well-predictable networks. | Suited for dynamic, variable networks. |

**22. What are the reasons for congestion in network layer?  [7M] [Dec/Jan202223] Understand CO2.**

Ans: The reasons for congestion in network layer are:

1. Too Much Traffic: Imagine a network as a road, and data packets as cars. When too many cars (data packets) try to use the road (network) at the same time, it gets congested. This often happens during busy times, like rush hour on a highway.

2. Traffic Spikes: Sometimes, the network has to handle sudden bursts of data, like when lots of people start streaming videos or downloading files all at once. This can overwhelm the network's capacity and cause congestion, just like a traffic jam during a big event

3. Network Bottlenecks: Think of the network as a pipeline. If some parts of the pipeline are narrow, they can slow down the flow of data. When too much data passes through these narrow points, congestion can occur.

4. Mistakes in Network Setup: Sometimes, network devices like routers and switches are not set up correctly. It's like having a traffic cop directing cars the wrong way. Misconfigured devices can cause traffic to get mixed up or stuck, leading to congestion.

5. Data Loss: If data packets get lost because of errors or too much traffic, they have to be sent again. This is like missing your exit on the highway and having to turn around, causing delays and congestion.

6. Confused Directions: Networks use specific paths to send data. If these paths are not chosen wisely, it's like taking a longer route to your destination, causing extra traffic and congestion.

7. Network Structure: How the network is designed can affect congestion. For example, if too many devices connect to a single point in the network, it's like too many people trying to go through a small door at once, causing congestion.

8. Security Measures: Security systems like firewalls can inspect data before allowing it through. This inspection takes time and can slow down traffic, leading to congestion during high security situations.

9. Changing Traffic Patterns: Just as traffic on the road can change unexpectedly, network traffic can shift suddenly. This can confuse the network and lead to congestion.

10. Failing Roads (Links): In a network, links or connections can fail. When this happens, traffic needs to be rerouted, and these detours can lead to congestion.

11. Bad Drivers (Inefficient Protocols): Sometimes, the way data packets are handled isn't efficient. It's like drivers who don't follow the rules. Inefficient protocols can clog up the network.

12. Limited Parking (Buffer Space): Network devices have limited space to store data temporarily. If this space is full, it's like a parking lot being completely occupied. New data can't come in, and this leads to congestion.

**23.What are the problems with congestion control?  [7M] [Dec/Jan202223] Understand CO2**

Ans : The problems with congestion control are as follows :

1. Wasting Resources: Sometimes, congestion control can be too careful, causing network resources to be underused. This happens when it restricts traffic too much, leading to slower network performance.

2. Being Too Aggressive: On the other hand, overly aggressive congestion control can be unfair. Some data flows might get too much network capacity, while others get very little, causing problems.

3. Buffer Overload: Congestion control can sometimes make network buffers too big, leading to higher delays and jitter in the network, which affects the user experience.

4. Complexity: The rules for congestion control can be quite complex, making it hard to design, implement, and fix problems with them.

5. Mixing Different Rules: Sometimes, different congestion control rules don't work well together, especially when networks with different rules interact. This can lead to inefficient network behaviour.

6. Handling Big Networks: In large networks, the coordination needed for congestion control can become a problem and lead to inefficiencies.

7. Delays in Feedback: Some congestion control rules depend on getting signals back from the network. Delays in receiving these signals can make congestion control less responsive.

8. Trouble with Real-time Apps: Some congestion control rules don't work well for real-time applications like video calls and online games because they can cause delays.

9. Changing Networks: Network conditions can change, and congestion control rules may not adapt quickly enough.

10. Security Concerns: In some cases, congestion control can be exploited by malicious users for attacks or network disruptions.

11. Inefficiency on Certain Networks: Some congestion control rules don't work well on networks with long delays or very high speeds.

12. Unintended Problems: Changes in congestion control can sometimes lead to new issues. For example, trying to make things fair might create new problems.

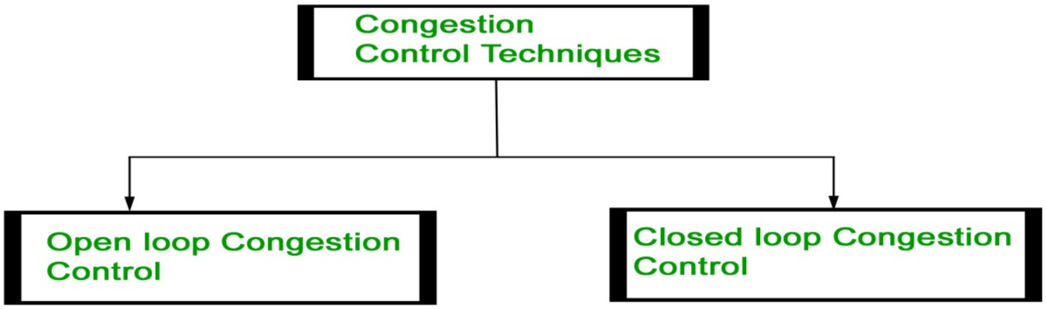
These are common challenges in the field of network engineering and congestion control, and researchers work on finding better solutions to these problems to make networks work more efficiently and fairly.

**24. Explain the general principles and prevention policies of congestion control. [7M] [Nov2019] Understand CO2**

Ans: The general principles and prevention policies of congestion control are :

General Principles of Congestion Control:

Congestion control refers to the techniques used to control or prevent congestion. Congestion control techniques can be broadly classified into two categories:


Congestion Control Techniques
Open loop Congestion Control
Closed loop Congestion Control

**Open Loop Congestion Control**
Open loop congestion control policies are applied to prevent congestion before it happens. The congestion control is handled either by the source or the destination.

**Policies adopted by open loop congestion control –**

1. **Retransmission Policy :**
   It is the policy in which retransmission of the packets are taken care of. If the sender feels that a sent packet is lost or corrupted, the packet needs to be retransmitted. This transmission may increase the congestion in the network.
   To prevent congestion, retransmission timers must be designed to prevent congestion and also able to optimize efficiency.

2. **Window Policy :**
   The type of window at the sender's side may also affect the congestion. Several packets in the Go-back-n window are re-sent, although some packets may be received successfully at the receiver side. This duplication may increase the congestion in the network and make it worse.
   Therefore, Selective repeat window should be adopted as it sends the specific packet that may

have been lost.

3. **Discarding Policy :**
   A good discarding policy adopted by the routers is that the routers may prevent congestion and at the same time partially discard the corrupted or less sensitive packages and also be able to maintain the quality of a message.
   In case of audio file transmission, routers can discard less sensitive packets to prevent congestion and also maintain the quality of the audio file.

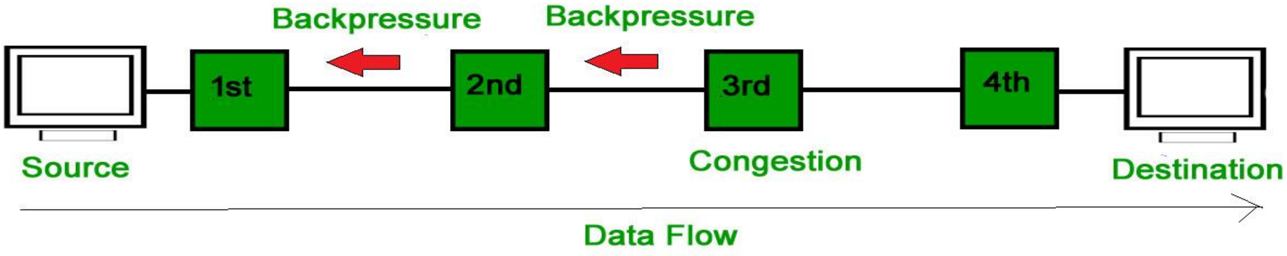
4. **Acknowledgment Policy :**
   Since acknowledgements are also the part of the load in the network, the acknowledgment policy imposed by the receiver may also affect congestion. Several approaches can be used to prevent congestion related to acknowledgment.
   The receiver should send acknowledgement for N packets rather than sending acknowledgement for a single packet. The receiver should send an acknowledgment only if it has to send a packet or a timer expires.

5. **Admission Policy :**
   In admission policy a mechanism should be used to prevent congestion. Switches in a flow should first check the resource requirement of a network flow before transmitting it further. If there is a chance of a congestion or there is a congestion in the network, router should deny establishing a virtual network connection to prevent further congestion.

All the above policies are adopted to prevent congestion before it happens in the network.

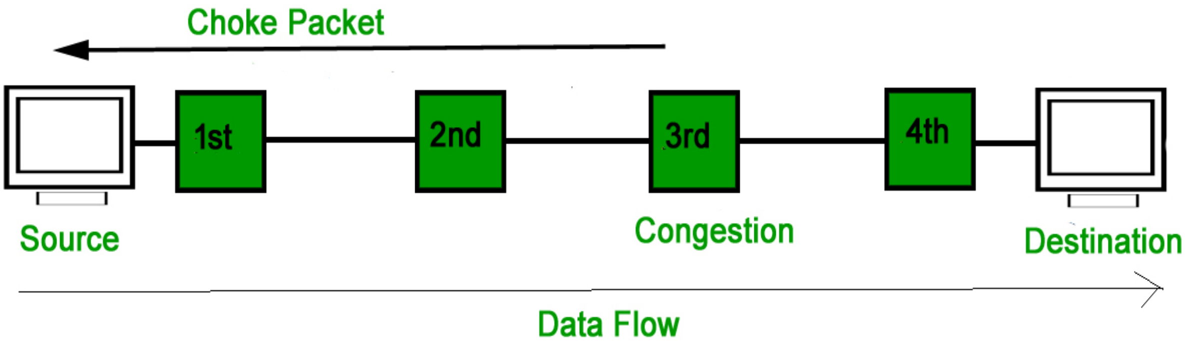
**Closed Loop Congestion Control**
Closed loop congestion control techniques are used to treat or alleviate congestion after it happens. Several techniques are used by different protocols; some of them are:

**1. Backpressure :**
Backpressure is a technique in which a congested node stops receiving packets from upstream node. This may cause the upstream node or nodes to become congested and reject receiving data from above nodes. Backpressure is a node-to-node congestion control technique that propagate in the opposite direction of data flow. The backpressure technique can be applied only to virtual circuit where each node has information of its above upstream node.



In above diagram the 3rd node is congested and stops receiving packets as a result 2nd node may be get congested due to slowing down of the output data flow. Similarly 1st node may get congested and inform the source to slow down.

**2. Choke Packet Technique :**
Choke packet technique is applicable to both virtual networks as well as datagram subnets. A choke packet is a packet sent by a node to the source to inform it of congestion. Each router monitors its resources and the utilization at each of its output lines. Whenever the resource utilization exceeds the threshold value which is set by the administrator, the router directly sends a choke packet to the source giving it a feedback to reduce the traffic. The intermediate nodes through which the packets has traveled are not warned about congestion.



**3. Implicit Signaling :**
In implicit signaling, there is no communication between the congested nodes and the source. The source guesses that there is congestion in a network. For example when sender sends several packets and there is no acknowledgment for a while, one assumption is that there is a congestion.

**4. Explicit Signaling :**
In explicit signaling, if a node experiences congestion it can explicitly sends a packet to the source or destination to inform about congestion. The difference between choke packet and explicit signaling is that the signal is included in the packets that carry data rather than creating a different packet as in case of choke packet technique.
Explicit signaling can occur in either forward or backward direction.

- **Forward Signaling :** In forward signaling, a signal is sent in the direction of the congestion. The destination is warned about congestion. The receiver in this case adopt policies to prevent further congestion.
- **Backward Signaling :** In backward signaling, a signal is sent in the opposite direction of the congestion. The source is warned about congestion and it needs to slow down.

**25. What is the role of different layers in controlling the congestion? Describe congestion control in datagram subnets. [7M] [Nov2019] Understand CO2**

Ans : Role of Different Layers in Controlling Congestion:

In networking, different layers work together to control congestion and ensure that data flows smoothly across the network. These layers are often referred to as the OSI (Open Systems Interconnection) model, which consists of seven layers. Let's focus on the relevant layers for congestion control:

1. Network Layer (Layer 3): The network layer is responsible for routing packets from the source to the destination. It plays a crucial role in managing congestion by determining the most efficient path for data to travel. When congestion occurs, routers in the network layer may reroute traffic to less congested paths to alleviate the problem.

2. Transport Layer (Layer 4): The transport layer manages end-to-end communication between devices. It includes protocols like TCP (Transmission Control Protocol) and UDP (User Datagram Protocol). TCP, in particular, employs congestion control mechanisms. It adjusts the sending rate of data to prevent network congestion. It uses feedback signals like packet acknowledgments and detects congestion based on packet loss.

3. Data Link Layer (Layer 2): The data link layer focuses on the physical and logical link between two devices. It plays a role in congestion control by ensuring that data is transmitted efficiently over the link, but it doesn't directly manage congestion in the same way as the higher layers.

Congestion Control in Datagram Subnets:

In datagram subnets, which are essentially networks that use the best-efforts delivery approach (like the Internet), congestion control becomes critical to ensure smooth data transmission. Here's how congestion control works in datagram subnets:

BestEffort Delivery: Datagram subnets, including the Internet, follow a best-efforts delivery model. This means that they don't guarantee a certain level of service but do their best to deliver packets.

RouterBased Congestion Control: Routers within the network are vital in managing congestion. They monitor network traffic and buffer incoming packets. If a router's buffers become too full, it may start dropping packets, which serves as a signal to the sender that the network is congested.

End-to-end Congestion Control: End-to-end congestion control is primarily managed by the transport layer, particularly with TCP. When TCP detects packet loss due to congestion (signaled by dropped packets), it slows down the sending rate to ease network congestion. It gradually increases the rate when congestion subsides.

Packet Discard Policies: Routers in datagram subnets often use policies like Random Early Detection (RED) to manage congestion. RED drops packets randomly when congestion starts, discouraging senders from overwhelming the network with more traffic.

Fairness: Congestion control in datagram subnets aims to be fair to all users sharing the network. This ensures that no single user or application monopolizes network resources, maintaining fairness in data transmission.

**26. Explain the leaky bucket and token bucket algorithm. [7M] [Dec/Jan202223] Understand CO2**

Ans : Leaky Bucket Algorithm:

Imagine a bucket with a leak at the bottom. The leaky bucket algorithm is like this bucket, and it's used to control the rate at which data is sent from a source to a destination. Here's how it works:

1. Bucket and Water: Think of the bucket as a buffer, and data packets as drops of water. The bucket has a fixed capacity, and it can hold a limited amount of water at any time.

2. Incoming Data: Data packets arrive at the source, and they're added to the bucket. If the bucket is already full, any excess packets are discarded.

3. Leak Rate: There's a hole at the bottom of the bucket, and it leaks water at a constant rate. This rate represents the maximum rate at which data can be sent out of the bucket.

4. Smooth Output: The leaky bucket ensures a smooth and controlled output of data. Even if packets arrive in bursts, the bucket's leak rate regulates the flow so that it doesn't overwhelm the network.

5. Rate Limiting: If data arrives too quickly, and the bucket starts to fill up, the excess packets are discarded. This helps prevent congestion by limiting the rate of outgoing data.

The leaky bucket algorithm is commonly used in network traffic shaping to ensure a consistent and controlled data flow, which is essential for quality of service and network management.

Token Bucket Algorithm:

The token bucket algorithm is similar to the leaky bucket, but it uses a different analogy involving tokens. Here's how it works:

1. Token Bucket: Imagine a bucket that holds tokens. Tokens are like permission slips to send data. The bucket has a maximum capacity for tokens.

2. Token Generation: Tokens are generated at a fixed rate and added to the bucket over time. The rate at which tokens are generated represents the allowed sending rate.

3. Data Packets: When data packets arrive, they need a token to be sent. If there are enough tokens in the bucket, the packet gets a token and is allowed to be sent.

4. Rate Control: The token bucket controls the sending rate by regulating the number of tokens in the bucket. If there are no tokens available, the packet has to wait until one becomes available.

5. Congestion Control: This algorithm helps prevent congestion because it ensures that data is sent at a controlled rate, and when the bucket is empty (no tokens), no additional data can be sent until tokens become available.

In essence, the token bucket algorithm uses tokens to control the rate at which data is sent, making it a useful tool for network traffic policing, ensuring that data is transmitted according to a specified rate, and preventing network congestion.

**27. Explain the need and the process of Network Address Translation. [7M] [Supply July – 2023] Analyze CO2**

Ans : Network Address Translation (NAT) :

Need for NAT:

1. Address Shortage: Imagine you have a limited number of unique street addresses, and everyone in your apartment complex or neighborhood needs one. NAT is like the mailroom that allows all residents to share a single unique street address (public IP) when sending or receiving mail (data) from the outside world (the internet).

2. Security Barrier: The mailroom (NAT) also acts as a security checkpoint. It keeps incoming mail (data) from directly reaching your apartment (devices) without proper identification. This adds a basic level of security because it hides your exact apartment number from outsiders.

3. Efficiency: NAT ensures that mail (data) from multiple apartments (devices) doesn't get mixed up. It assigns each piece of mail a temporary apartment number (port number) so that when it arrives, it can be directed to the correct apartment (device).

Process of NAT:

1. Private vs. Public Addresses: In your apartment complex (local network), everyone has a private apartment number (private IP) that isn't unique outside your complex. The apartment building itself has a unique street address (public IP) for interactions with the outside world (internet).

2. Translation Table: The mailroom (NAT device) maintains a logbook (translation table) that keeps track of which private apartment (device) sent what piece of mail (data). It records the apartment's number and the temporary apartment number (port) assigned to the mail.

3. Outgoing Mail (Source NAT): When an apartment (device) sends mail (data) out to the world, the mailroom changes the return address on the envelope (data packet). It replaces the apartment's number (private IP) with the building's street address (public IP) and assigns a unique temporary apartment number (port) to each piece of outgoing mail. This is like Source Network Address Translation (SNAT).

4. Incoming Mail (Destination NAT): When the response arrives, it's addressed to the building's street address (public IP). The mailroom checks its logbook (translation table) to see which apartment (device) was expecting this piece of mail. It forwards the mail to the right apartment, using the temporary apartment number (port) to ensure it gets to the correct recipient. This is like Destination Network Address Translation (DNAT).

5. Port Numbers: Port numbers are like apartment room numbers. They help NAT distinguish which piece of mail (data) belongs to which apartment (device). This way, multiple apartments (devices) can share the same street address (public IP) without their mail (data) getting mixed up.

6. Dynamic and Static NAT: Think of Dynamic NAT like sharing the mailroom's street address among all apartments, with the mailroom managing who gets to use it at any given time. Static NAT is when specific apartments (devices) have their own dedicated mail slots (public IPs) and the mailroom knows exactly where each piece of mail should go.

In essence, NAT acts as a mailroom for your devices, allowing multiple devices to share a single unique street address (public IP), enhancing security, and ensuring efficient delivery of data to the correct device.

**28. A university has 264 LANs in it. Each LAN contains 50 hosts. Suppose the university has one class B address. Design an appropriate subnet addressing scheme. [7M][Supply July – 2023] Analyze CO2**

Ans : To design an appropriate subnet addressing scheme for a university with 264 LANs, each containing 50 hosts, using a single class B address, you'll need to make efficient use of IP address space. Here's how you can do it :

Step 1: Determine the Number of Hosts per LAN

You have 264 LANs, and each LAN contains 50 hosts. So, you need to accommodate 50 hosts per LAN. Since 50 is not a power of 2, we'll round it up to the nearest power of 2, which is 64 (2^6). This means you need 6 bits for host addresses within each LAN.

Step 2: Determine the Number of Subnets

To accommodate 264 LANs, you need a minimum of 9 bits for subnet addresses. This is because 2^9 = 512, which is the smallest power of 2 greater than 264.

Step 3: Create the Subnet Addressing Scheme

Now, you have a class B address which, by default, has a subnet mask of 255.255.0.0. You need to borrow some bits from the host portion to create subnets. In this case, you need to borrow 9 bits for subnets and 6 bits for hosts within each subnet.

The subnet mask would look like this: 255.255.11111111.00000000, which can be written as 255.255.255.0 (in decimal form) or /24 (in CIDR notation).

Here's how you can allocate your class B address space:

Subnet 1: 172.16.1.0/24  This subnet can accommodate 256 addresses (50 hosts + 1 for the subnet address and 1 for the broadcast address).

Subnet 2: 172.16.2.0/24

Subnet 3: 172.16.3.0/24

...

Subnet 264: 172.16.264.0/24

Each of these subnets can support up to 256 IP addresses, but you only need 50 for hosts, so you have some room for growth within each LAN.

Step 4: Implementation

You would need to configure the routers and switches within the university network to use this subnet addressing scheme. Each LAN would be assigned one of these subnets, and the devices within that LAN would be configured with IP addresses from the corresponding subnet.

In summary, this subnet addressing scheme allows the university to efficiently manage 264 LANs, each with 50 hosts, using a single class B address by borrowing 9 bits for subnets and 6 bits for hosts within each subnet. This approach ensures that there are enough subnets and addresses for the university's needs while maintaining room for future growth.

**29. How Networks differ, how networks can be connected. [7M] [Nov2019] Analyze CO2**

Ans : How Networks Differ:

Networks can differ in several ways, which can be understood by thinking of them like different types of roads:

1. Size and Scope:
   Local Area Network (LAN): Think of a LAN like a neighborhood road. It's small and connects devices within a limited area, like a home, office, or campus.
   Wide Area Network (WAN): A WAN is like a city's main highways. It connects LANs over a larger geographical area, often spanning cities, states, or even countries.

2. Technology and Protocols:
   Ethernet: Imagine Ethernet like a standard two-lane road. It's commonly used within LANs.
   Internet: The internet is like a massive highway system, connecting different cities and countries, and it uses various protocols like TCP/IP.

3. Ownership and Control:

Private Network: This is like a privately owned road within a factory or office complex, controlled by a single organization.

Public Network: Think of a public network like government-managed roads. They are available for public use, and anyone can access them, like the internet.

4. Topology:

Star Topology: Imagine this as a hub-and-spoke road network. All devices are connected to a central hub.

Mesh Topology: Picture a fully interconnected grid of roads. In a network, this means every device connects to every other device.

5. Purposes:

Corporate Network: Think of this like a business district with private roads, connecting offices and departments.

Academic Network: An academic network is like a university campus network, connecting different buildings and facilities.

How Networks Can Be Connected:

1. Physical Connection:

Wired: Networks can be connected using physical cables, similar to roads. Ethernet cables are like the roads for data, ensuring reliable and fast connections.

Wireless: Wireless networks are like invisible airways for data. Devices communicate using radio signals, similar to how cars on different roads can't see each other but can communicate.

2. Router and Switches:

Router: Think of a router like a traffic cop directing data to the right path. It connects different networks (like LANs) and manages traffic between them.

Switch: A switch is like an intersection on a road network, directing data within a single network to the right device.

3. Internet Connection:

The internet is like a giant highway system. You connect to the internet via an Internet Service Provider (ISP), which is like the main highway entrance. The ISP routes your data to its destination, much like a long-distance road trip.

4. Tunneling:

Tunneling is like creating a secret passage beneath the roads. It's used to securely transmit data across a public network, such as the internet, as if it were traveling through a private network.

**30. What is internet working? Explain about tunnelling. [7M] [Dec/Jan-2022-23]Understand CO2**

Ans:
Internetworking:

Imagine the internet as a vast network of networks, where each network can be a local network (like your home Wi-Fi) or a larger network (like a company's network). Internetworking is the art of connecting these different networks so that they can communicate and share data.

Here's how it works:

1. Protocols: Just like roads have rules and signs, networks have protocols (like TCP/IP) that determine how data should be packaged and sent between different networks.

2. Routers: Think of routers as the traffic cops of the internet. They sit at the intersections between networks and make sure data takes the right path to reach its destination. Routers read the addresses on data packets and decide where to send them next.

3. Addresses: Every device connected to the internet has a unique address, just like every house has a street address. This address helps routers know where to send data. For example, an IP address is like a device's "location" on the internet.

4. Subnets: Networks are often divided into smaller parts called subnets. Subnets are like neighborhoods with their own local roads. Routers use subnet information to navigate data within larger networks.

5. Firewalls: Firewalls are like security gates on roads. They ensure that only allowed data can enter or exit a network. Firewalls help protect networks from unauthorized access or malicious data.

Tunneling:

Tunneling is a clever way to send data securely through an insecure or public network, like sending secret messages through a busy street without anyone reading them. Here's how it works:

1. Encapsulation: Imagine you have a secret letter (your data) that you want to send through a crowded street (the internet). With tunneling, you put this letter in a special, indestructible envelope (the tunnel).

2. Tunnel Entrance: Next, you find a secret entrance (the tunnel entrance) to the street. This entrance is like a hidden pathway that only you and your intended recipient know about.

3. Safe Passage: You drop your envelope into this secret entrance, and it travels through the hidden tunnel (the secure pathway) inside the busy street. It's protected from prying eyes and any potential dangers on the street.

4. Exit Point: At the other end of the street, your envelope comes out of another secret exit (the tunnel exit). Only your intended recipient knows where this exit is.

5. Decryption: Your recipient receives the envelope, opens it, and reads your letter. Since the envelope is indestructible, the letter inside is safe and intact.

**31. Discuss different internet control protocols.**

**Ans) Types of Internet Protocol**

Internet Protocols are of different types having different uses. These are mentioned below:

1. TCP/IP(Transmission Control Protocol/ Internet Protocol)
2. SMTP(Simple Mail Transfer Protocol)
3. PPP(Point-to-Point Protocol)
4. FTP (File Transfer Protocol)
5. SFTP(Secure File Transfer Protocol)
6. HTTP(Hyper Text Transfer Protocol)
7. HTTPS(Hyper Text Transfer Protocol Secure)
8. TELNET(Terminal Network)
9. POP3(Post Office Protocol 3)
10. IPv4
11. IPv6
12. ICMP
13. UDP
14. IMAP
15. SSH
16. Gopher

**1. TCP/IP(Transmission Control Protocol/ Internet Protocol)**

These are a set of standard rules that allows different types of computers to communicate with each other. The IP protocol ensures that each computer that is connected to the Internet is having a specific serial number called the IP address. TCP specifies how data is exchanged over the internet and how it should be broken into IP packets. It also makes sure that the packets have information about the source of the message data, the destination of the message data, the sequence in which the message data should be re-assembled, and checks if the message has been sent correctly to the specific destination. The TCP is also known as a connection-oriented protocol.

For more details, please refer TCP/IP Model article.

**2. SMTP(Simple Mail Transfer Protocol)**

These protocols are important for sending and distributing outgoing emails. This protocol uses the header of the mail to get the email id of the receiver and enters the mail into the queue of outgoing mail. And as soon as it delivers the mail to the receiving email id, it removes the email from the outgoing list. The message or the electronic mail may consider the text, video, image, etc. It helps in setting up some communication server rules.

**3. PPP(Point-to-Point Protocol)**

It is a communication protocol that is used to create a direct connection between two communicating devices. This protocol defines the rules using which two devices will authenticate with each other and exchange information with each other. For example, A user connects his PC to the server of an Internet Service Provider and also uses PPP. Similarly, for connecting two routers for direct communication it uses PPP.

**4. FTP (File Transfer Protocol)**

This protocol is used for transferring files from one system to the other. This works on a client-server model. When a machine requests for file transfer from another machine, the FTO sets up a connection between the two and authenticates each other using their ID and Password. And, the desired file transfer takes place between the machines.

**5. SFTP(Secure File Transfer Protocol)**

SFTP which is also known as SSH FTP refers to File Transfer Protocol (FTP) over Secure Shell (SSH) as it encrypts both commands and data while in transmission. SFTP acts as an extension to SSH and encrypts files and data then sends them over a secure shell data stream. This protocol is used to remotely connect to other systems while executing commands from the command line.

**6. HTTP(Hyper Text Transfer Protocol)**

This protocol is used to transfer hypertexts over the internet and it is defined by the www(world wide web) for information transfer. This protocol defines how the information needs to be formatted and transmitted. And, it also defines the various actions the web browsers should take in response to the calls made to access a particular web page. Whenever a user opens their web browser, the user will indirectly use HTTP as this is the protocol that is being used to share text, images, and other multimedia files on the World Wide Web.

Note: Hypertext refers to the special format of the text that can contain links to other texts.

**7. HTTPS(HyperText Transfer Protocol Secure)**

HTTPS is an extension of the Hypertext Transfer Protocol (HTTP). It is used for secure communication over a computer network with the SSL/TLS protocol for encryption and authentication. So, generally, a website has an HTTP protocol but if the website is such that it receives some sensitive information such as credit card details, debit card details, OTP, etc then it requires an SSL certificate installed to make the website more secure. So, before entering any sensitive information on a website, we should check if the link is HTTPS or not. If it is not HTTPS then it may not be secure enough to enter sensitive information.

**8. TELNET(Terminal Network)**

TELNET is a standard TCP/IP protocol used for virtual terminal service given by ISO. This enables one local machine to connect with another. The computer which is being connected is called a remote computer and which is connecting is called the local computer. TELNET operation lets us display anything being performed on the remote computer in the local computer. This operates on the client/server principle. The local computer uses the telnet client program whereas the remote computer uses the telnet server program.

**9. POP3(Post Office Protocol 3)**

POP3 stands for Post Office Protocol version 3. It has two Message Access Agents (MAAs) where one is client MAA (Message Access Agent) and another is server MAA(Message Access Agent) for accessing the messages from the mailbox. This protocol helps us to retrieve and manage emails from the mailbox on the receiver mail server to the receiver's computer. This is implied between the receiver and the receiver mail server. It can also be called a one-way client-server protocol. The POP3 WORKS ON THE 2 PORTS I.E. PORT 110 AND PORT 995.

**10. IPv4**

The fourth and initially widely used version of the Internet Protocol is called IPv4 (Internet Protocol version 4). It is the most popular version of the Internet Protocol and is in charge of distributing data packets throughout the network. Maximum unique addresses for IPv4 are 4,294,967,296 (232), which are possible due to the use of 32-bit addresses. The network address and the host address are the two components of each address. The host address identifies a particular device within the network, whereas the network address identifies the network to which the host belongs. In the "dotted decimal" notation, which is the standard for IPv4 addresses, each octet (8 bits) of the address is represented by its decimal value and separated by a dot (e.g. 192.168.1.1).

**11. IPv6**

The most recent version of the Internet Protocol, IPv6, was created to address the IPv4 protocol's drawbacks. A maximum of 4.3 billion unique addresses are possible with IPv4's 32-bit addresses. Contrarily, IPv6 uses 128-bit addresses, which enable a significantly greater number of unique addresses. This is significant because IPv4 addresses were running out and there are an increasing number of devices that require internet access. Additionally, IPv6 offers enhanced security features like integrated authentication and encryption as well as better support for mobile devices. IPv6 support has spread among websites and internet service providers, and it is anticipated to gradually displace IPv4 as the main internet protocol.

For more details, please refer Differences between IPv4 and IPv6 article.

**12. ICMP**

ICMP (Internet Control Message Protocol) is a network protocol that is used to send error messages and operational information about network conditions. It is an integral part of the Internet Protocol (IP) suite and is used to help diagnose and troubleshoot issues with network connectivity. ICMP messages are typically generated by network devices, such as routers, in response to errors or exceptional conditions encountered in forwarding a datagram. Some examples of ICMP messages include:

- Echo Request and Echo Reply (ping)
- Destination Unreachable
- Time Exceeded
- Redirect

ICMP can also be used by network management tools to test the reachability of a host and measure the round-trip time for packets to travel from the source to the destination and back. It should be noted that ICMP is not a secure protocol, it can be used in some types of network attacks like DDoS amplification.

**13. UDP**

UDP (User Datagram Protocol) is a connectionless, unreliable transport layer protocol. Unlike TCP, it does not establish a reliable connection between devices before transmitting data, and it does not guarantee that data packets will be received in the order they were sent or that they will be received at all. Instead, UDP simply sends packets of data to a destination without any error checking or flow control. UDP is typically used for real-time applications such as streaming video and audio, online gaming, and VoIP (Voice over Internet Protocol) where a small amount of lost data is acceptable and low latency is important. UDP is faster than TCP because it has less overhead. It doesn’t need to establish a connection, so it can send data packets immediately. It also doesn’t need to wait for confirmation that the data was received before sending more, so it can transmit data at a higher rate.

**14. IMAP**

IMAP (Internet Message Access Protocol) is a protocol used for retrieving emails from a mail server. It allows users to access and manage their emails on the server, rather than downloading them to a local device. This means that the user can access their emails from multiple devices and the emails will be synced across all devices. IMAP is more flexible than POP3 (Post Office Protocol version 3) as it allows users to access and organize their emails on the server, and also allows multiple users to access the same mailbox.

**15. SSH**

SSH (Secure Shell) is a protocol used for secure remote login and other secure network services. It provides a secure and encrypted way to remotely access and manage servers, network devices, and other computer systems. SSH uses public-key cryptography to authenticate the user and encrypt the data being transmitted, making it much more secure than traditional remote login protocols such as Telnet. SSH also allows for secure file transfers using the SCP (Secure Copy) and SFTP (Secure File Transfer Protocol) protocols. It is widely used in Unix-based operating systems and is also available for Windows. It is commonly used by system administrators, developers, and other technical users to remotely access and manage servers and other network devices.

### 16. Gopher

Gopher is a type of file retrieval protocol that provides downloadable files with some description for easy management, retrieving, and searching of files. All the files are arranged on a remote computer in a stratified manner. It is an old protocol and it is not much used nowadays.

### 32. Classify Internet, Intranet and Extranet with applications

**Ans)**

**Difference between Internet, Intranet and Extranet**

**1. Internet :**

The network formed by the co-operative interconnection of millions of computers, linked together is called Internet. Internet comprises of :

- **People :** People use and develop the network.
- **Resources :** A collection of resources that can be reached from those networks.
- **A setup for collaboration :** It includes the member of the research and educational committees worldwide.

**2. Intranet :**

It is an internal private network built within an organization using Internet and World Wide Web standards and products that allows employees of an organization to gain access to corporate information.

**3. Extranet :**

It is the type of network that allows users from outside to access the Intranet of an organization.

**Difference between Internet, Intranet and Extranet :**

| Point of difference | Internet | Intranet | Extranet |
|---|---|---|---|
| Accessibility of network | Public | Private | Private |

| | | | |
|---|---|---|---|
| Availability | Global system. | Specific to an organization. | To share information with suppliers and vendors it makes the use of public network. |
| Coverage | All over the world. | Restricted area upto an organization. | Restricted area upto an organization and some of its stakeholders or so. |
| Accessibility of content | It is accessible to everyone connected. | It is accessible only to the members of organization. | Accessible only to the members of organization and external members with logins. |
| No. of computers connected | It is largest in number of connected devices. | The minimal number of devices are connected. | The connected devices are more comparable with Intranet. |
| Owner | No one. | Single organization. | Single/ Multiple organization. |
| Purpose of the network | It’s purpose is to share information throughout the world. | It’s purpose is to share information throughout the organization. | It’s purpose is to share information between members and external, members. |
| Security | It is dependent on the user of the device connected to network. | It is enforced via firewall. | It is enforced via firewall that separates internet and extranet. |
| Users | General public. | Employees of the organization. | Employees of the organization which are connected. |

| | | | |
|---|---|---|---|
| Policies behind setup | There is no hard and fast rule for policies. | Policies of the organization are imposed. | Policies of the organization are imposed. |
| Maintenance | It is maintained by ISP. | It is maintained by CIO. HR or communication department of an organization. | It is maintained by CIO. HR or communication department of an organization. |
| Economical | It is more economical to use. | It is less economical. | It is also less economical. |
| Relation | It is the network of networks. | It is derived from Internet. | It is derived from Intranet. |
| Example | What we are normally using is internet. | WIPRO using internal network for its business operations. | DELL and Intel using network for its business operations. |

**33. Explain Internet control protocols- ICMP, ARP, DHCP.**

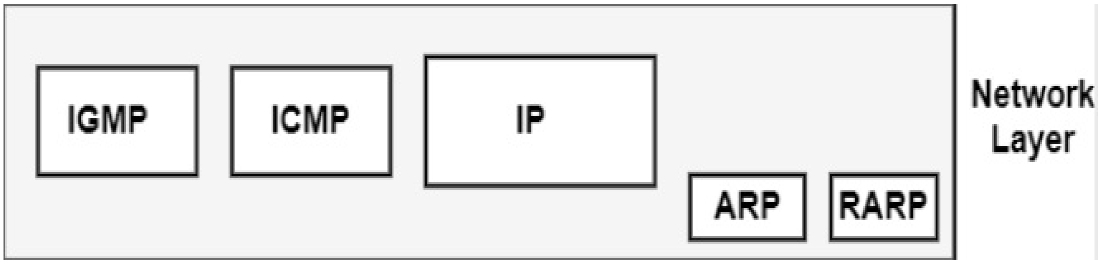
**Ans) ICMP Protocol :**

The ICMP represents Internet Control Message Protocol. It is a network layer protocol. It can be used for error handling in the network layer, and it is generally used on network devices, including routers. IP Protocol is a best-effect delivery service that delivers a datagram from its original source to its final destination. It has two deficiencies−

- Lack of Error Control
- Lack of assistance mechanisms

IP protocol also lacks a structure for host and management queries. A host needs to resolve if a router or another host is alive, and sometimes a network manager needs information from another host or router.

ICMP has been created to compensate for these deficiencies. It is a partner to the IP protocol.



ICMP is a network layer protocol. But, its messages are not passed directly to the data link layer. Instead, the messages are first encapsulated inside the IP datagrams before going to the lower layer.

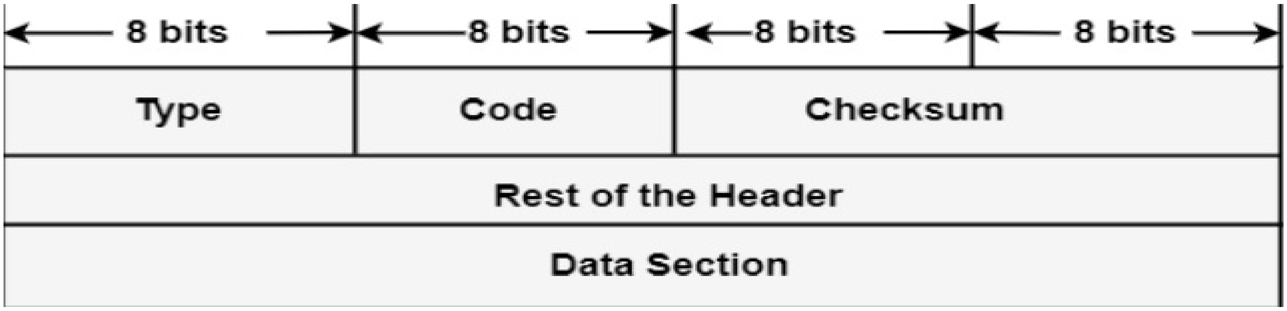
The cost of the protocol field in the IP datagram is I, to indicate that IP data is an ICMP message.

The error reporting messages report issues that a router or a host (destination) may encounter when it phases an IP packet.

The query messages, which appear in pairs, help a host or a network manager to get specific data from a router or another host.

ICMP Message Format

AN ICMP message includes an 8-byte header and a variable size data format.

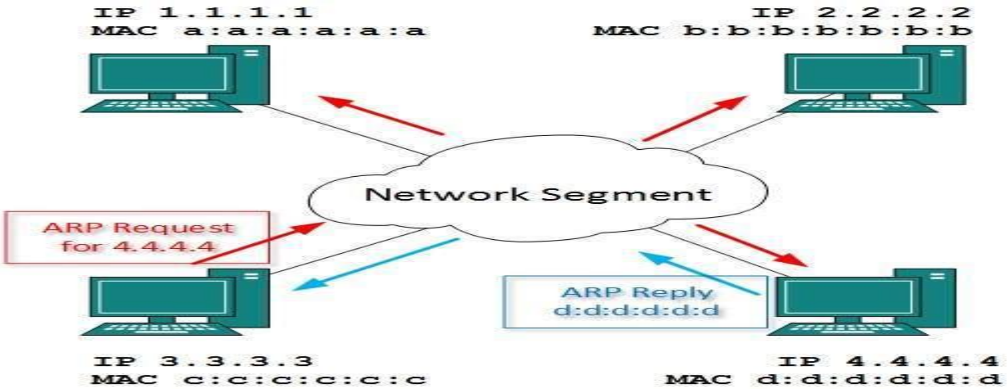


- **Type:** It is an 8-bit field. It represents the ICMP message type. The values area from 0 to 127 are described for ICMPv6, and the values from 128 to 255 are the data messages.
- **Code:** It is an 8-bit field that represents the subtype of the ICMP message.
- **Checksum:** It is a 16-bit field to recognize whether the error exists in the message or not.

Address Resolution Protocol(ARP):

While communicating, a host needs Layer-2 (MAC) address of the destination machine which belongs to the same broadcast domain or network. A MAC address is physically burnt into the Network Interface Card (NIC) of a machine and it never changes.

On the other hand, IP address on the public domain is rarely changed. If the NIC is changed in case of some fault, the MAC address also changes. This way, for Layer-2 communication to take place, a mapping between the two is required.



To know the MAC address of remote host on a broadcast domain, a computer wishing to initiate communication sends out an ARP broadcast message asking, “Who has this IP address?” Because it is a broadcast, all hosts on the network segment (broadcast domain) receive this packet and process it. ARP packet contains the IP address of destination host, the sending host wishes to talk to. When a host receives an ARP packet destined to it, it replies back with its own MAC address.

Once the host gets destination MAC address, it can communicate with remote host using Layer-2 link protocol. This MAC to IP mapping is saved into ARP cache of both sending and receiving hosts. Next time, if they require to communicate, they can directly refer to their respective ARP cache.

Reverse ARP is a mechanism where host knows the MAC address of remote host but requires to know IP address to communicate.

DHCP(Dynamic Host Configuration Protocol):

The Dynamic Host Configuration Protocol is a network protocol for Internet Protocol (IP) networks that assign IP addresses and other communication settings to devices connected to the network using a client-server architecture.

The technology is made up of two network components: a network DHCP server that is centrally deployed and the client instances of the protocol stack on each computer or device that eliminate the necessity for manually configuring the network devices. When one client first connects to the network, it uses the DHCP protocol to request a set of settings from the DHCP server.

DHCP is a client/server protocol that automatically assigns an IP address and other configuration information to an Internet Protocol (IP) host, such as the subnet mask and default gateway. When using DHCP, the server uses port 67 and the client uses port 68.

When a computer is connected into a different location on the network, DHCP allows a network administrator to oversee and distribute IP addresses from a central location, and it immediately transmits a new Internet Protocol (IP) address.

DHCP is an application layer protocol that provides −

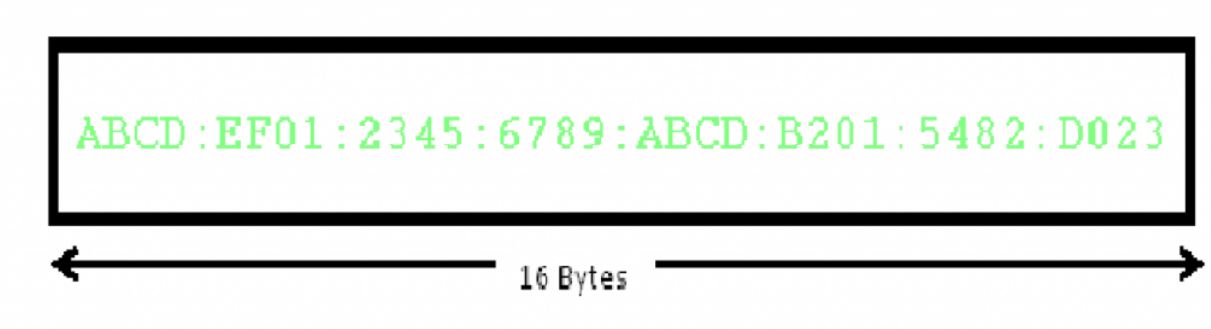
- Subnet Mask
- Router Address
- IP Address

DHCP may be used on a variety of networks, from small home networks to big university networks and regional ISP networks. DHCP server capability is available on many routers and residential gateways.

**34) What are the motivation factors for IPV6? Explain about the IPV6 address structure**.
**Ans)** IPv6 was developed by Internet Engineering Task Force (IETF) to deal with the problem of IPv4 exhaustion. IPv6 is a 128-bits address having an address space of $2^{128}$, which is way bigger than IPv4. IPv6 use Hexa-Decimal format separated by colon (:) .

**Components in Address format :**

1. There are 8 groups and each group represents 2 Bytes (16-bits).
2. Each Hex-Digit is of 4 bits (1 nibble)
3. Delimiter used – colon (:)



**Need for IPv6:**

The Main reason of IPv6 was the address depletion as the need for electronic devices rose quickly when Internet Of Things (IOT) came into picture after the 1980s & other reasons are related to the slowness of the process due to some unnecessary processing, the need for new options, support for multimedia, and the desperate need for security. IPv6 protocol responds to the above issues using the following main changes in the protocol:

**1. Large address space**

An IPv6 address is 128 bits long .compared with the 32 bit address of IPv4, this is a huge(2 raised 96 times) increases in the address space.

**2. Better header format**

IPv6 uses a new header format in which options are separated from the base header and inserted, when needed, between the base header and the upper layer data . This simplifies and speeds up the routing process because most of the options do not need to be checked by routers.

**3. New options**

IPv6 has new options to allow for additional functionalities.

**4. Allowance for extension**

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

**5. Support for resource allocation**

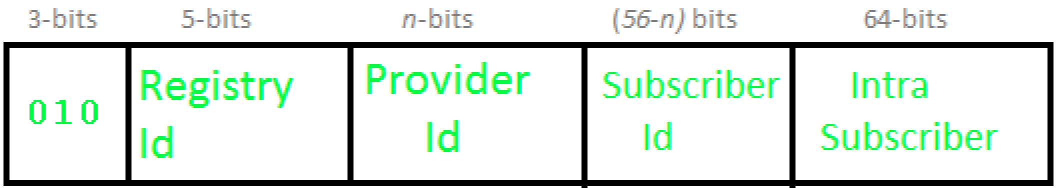
In IPv6,the type of service field has been removed, but two new fields , traffic class and flow label have been added to enables the source to request special handling of the packet . this mechanism can be used to support traffic such as real-time audio and video.

**6. Support for more security**

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

In IPv6 representation, we have three addressing methods :

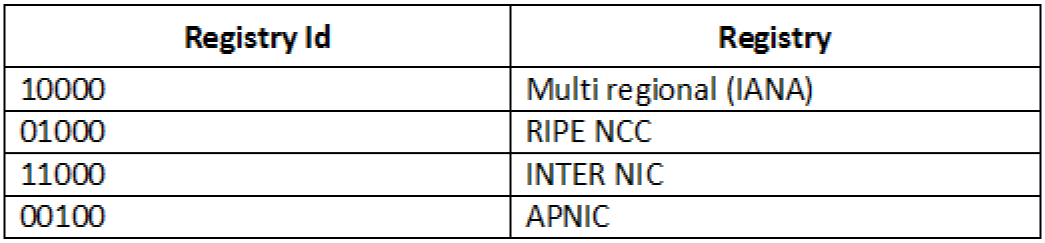
- Unicast
- Multicast
- Anycast
  - **Provider-based Unicast address :**
    These are used for global communication.



| 3-bits | 5-bits | n-bits | (56-n) bits | 64-bits |
|---|---|---|---|---|
| 010 | Registry Id | Provider Id | Subscriber Id | Intra Subscriber |

- The First 3 bits identify it as of this type.

  **Registry Id (5-bits):** Registry Id identifies the region to which it belongs. Out of 32 (i.e. 2^5), only 4 registry IDs are being used.

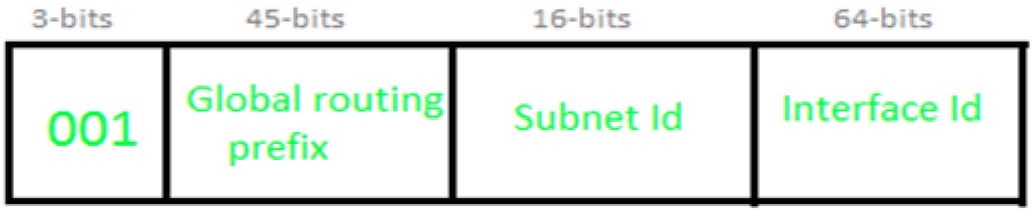


| Registry Id | Registry |
|---|---|
| 10000 | Multi regional (IANA) |
| 01000 | RIPE NCC |
| 11000 | INTER NIC |
| 00100 | APNIC |

- **Provider Id:** Depending on the number of service providers that operate under a region, certain bits will be allocated to the Provider Id field. This field need not be fixed. Let’s say if Provider Id = 10 bits then Subscriber Id will be 56 – 10 = 46 bits.

  **Subscriber Id:** After Provider Id is fixed, the remaining part can be used by ISP as a normal IP address.

  **Intra Subscriber:** This part can be modified as per the need of the organization that is using the service.
- **Geography based Unicast address :**



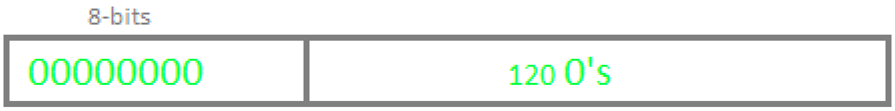
| 3-bits | 45-bits | 16-bits | 64-bits |
|---|---|---|---|
| 001 | Global routing prefix | Subnet Id | Interface Id |

- **Global routing prefix:** Global routing prefix contains all the details of Latitude and Longitude. As of now, it is not being used. In Geography-based Unicast address routing will be based on location.

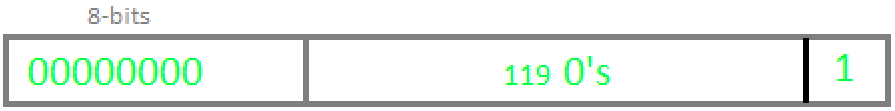
  **Interface Id:** In IPv6, instead of using Host Id, we use the term Interface Id.

  **Some special addresses:**

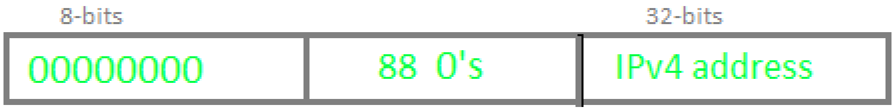
  **Unspecified**



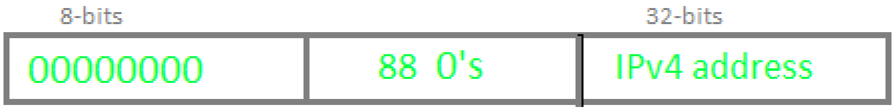
**Loopback**



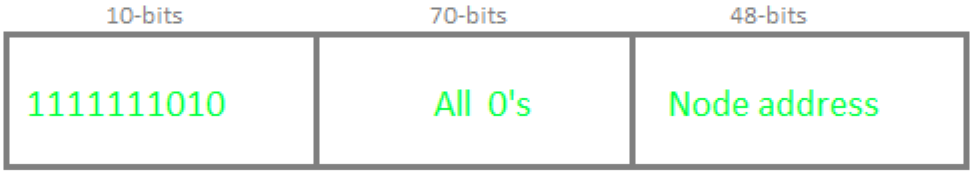
- **IPv4 Compatible**



- **IPv4 mapped**

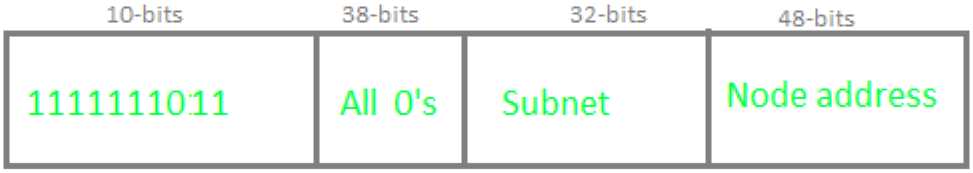


- Local Unicast Addresses :
  These are of two types: **Link-local** and **Site-Local**
- **1. Link-local address:**



- A link-local address is used for addressing a single link. It can also be used to communicate with nodes on the same link. The link-local address always begins with 1111111010 (i.e. FE80). The router will not forward any packet with Link-local address.

- **2. Site local address:**



| 10-bits | 38-bits | 32-bits | 48-bits |
|---|---|---|---|
| 1111110:11 | All 0's | Subnet | Node address |

- Site local addresses are equivalent to a private IP address in IPv4. Likely, some address space is reserved, which can only be routed within an organization. The first 10-bits are set to 1111111011, which is why Site local addresses always begin with FEC0. The following 32 bits are Subnet IDs, which can be used to create a subnet within the organization. The node address is used to uniquely identify the link; therefore, we use a 48-bits MAC address here.

**Addressing methods**

**1. Unicast Address**

Unicast Address identifies a single network interface. A packet sent to a unicast address is delivered to the interface identified by that address.

**2. Multicast Address**

Multicast Address is used by multiple hosts, called as **groups**, acquires a multicast destination address. These hosts need not be geographically together. If any packet is sent to this multicast address, it will be distributed to all interfaces corresponding to that multicast address. And every node is configured in the same way. In simple words, one data packet is sent to multiple destinations simultaneously.

**3. Anycast Address**

Anycast Address is assigned to a group of interfaces. Any packet sent to an anycast address will be delivered to only one member interface (mostly nearest host possible).

**Advantages of IPv6 :**

**1. Realtime Data Transmission :** Realtime data transmission refers to the process of transmitting data in a very fast manner or **immediately**. Example : Live streaming services such as cricket matches, or other tournament that are streamed on web exactly as soon as it happens with a maximum delay of 5-6 seconds.

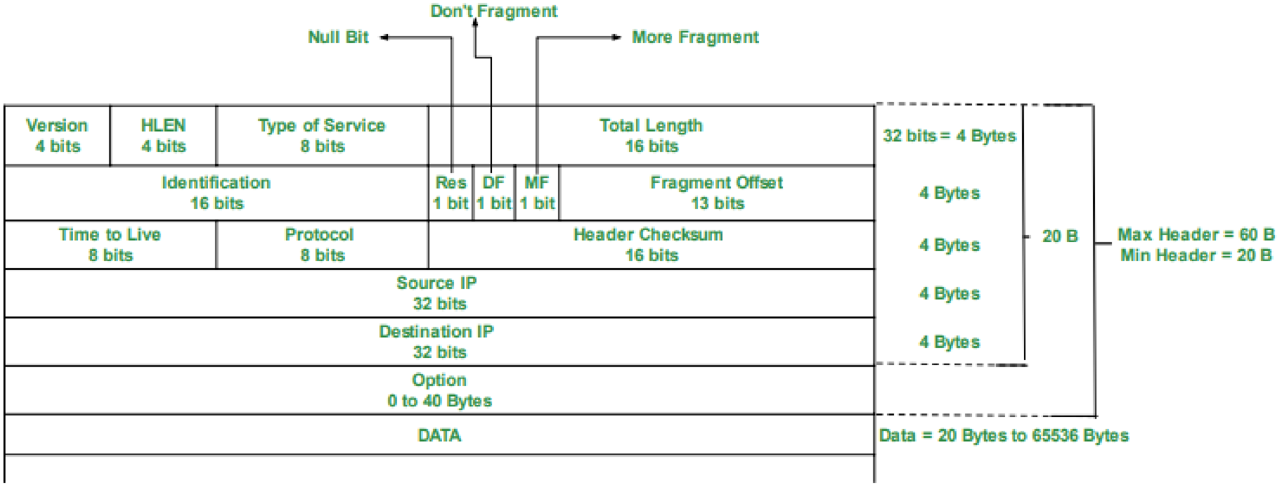
**2. IPv6 supports authentication:** Verifying that the data received by the receiver from the sender is exactly what the sender sent and came through the sender only not from any third party. Example : Matching the hash value of both the messages for verification is also done by IPv6.

**3. IPv6 performs Encryption:** Ipv6  can encrypt the message at network layer even if the protocols of application layer at user level didn't encrypt the message which is a major advantage as it takes care of encryption.

**4. Faster processing at Router:** Routers are able to process data packets of Ipv6 much faster due to smaller **Base header** of fixed size – 40 bytes which helps in decreasing processing time resulting in more efficient packet transmission. Whereas in Ipv4, we have to calculate the length of header which lies between 20-60 bytes.

**35.) Explain the IPV4 header format with neat sketch.**

**Ans) IPv4 Datagram Header** Size of the header is 20 to 60 bytes.



IPv4 Datagram Header

**VERSION:** Version of the IP protocol (4 bits), which is 4 for IPv4
**HLEN:** IP header length (4 bits), which is the number of 32 bit words in the header. The minimum value for this field is 5 and the maximum is 15.
**Type of service:** Low Delay, High Throughput, Reliability (8 bits)
**Total Length:** Length of header + Data (16 bits), which has a minimum value 20 bytes and the maximum is 65,535 bytes.
**Identification:** Unique Packet Id for identifying the group of fragments of a single IP datagram (16 bits)
**Flags:** 3 flags of 1 bit each : reserved bit (must be zero), do not fragment flag, more fragments flag (same order)
**Fragment Offset:** Represents the number of Data Bytes ahead of the particular fragment in the particular Datagram. Specified in terms of number of 8 bytes, which has the maximum value of 65,528 bytes.
**Time to live:** Datagram's lifetime (8 bits), It prevents the datagram to loop through the network by restricting the number of Hops taken by a Packet before delivering to the Destination.
**Protocol:** Name of the protocol to which the data is to be passed (8 bits)
**Header Checksum:** 16 bits header checksum for checking errors in the datagram header
**Source IP address:** 32 bits IP address of the sender
**Destination IP address:** 32 bits IP address of the receiver

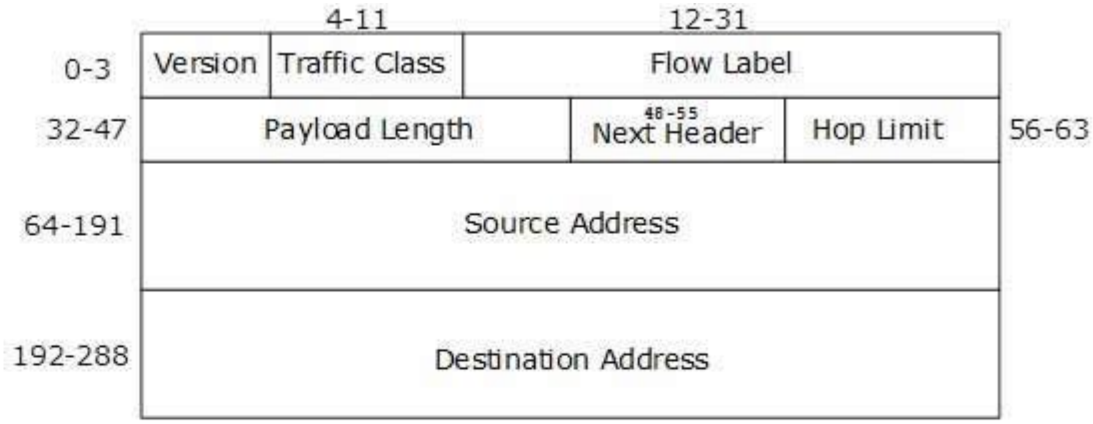
**Option:** Optional information such as source route, record route. Used by the Network administrator to check whether a path is working or not.

**36. Draw and explain IPV6 header.**

**[7M] [Supply-June/July-2022] Analysis CO2**

Ans) IPv6 lies in its header. An IPv6 address is 4 times larger than IPv4, but surprisingly, the header of an IPv6 address is only 2 times larger than that of IPv4. IPv6 headers have one Fixed Header and zero or more Optional (Extension) Headers. All the necessary information that is essential for a router is kept in the Fixed Header. The Extension Header contains optional information that helps routers to understand how to handle a packet/flow.

Fixed Header



[Image: IPv6 Fixed Header]

IPv6 fixed header is 40 bytes long and contains the following information.

| S.N. | Field & Description |
|---|---|
| 1 | **Version** (4-bits): It represents the version of Internet Protocol, i.e. 0110. |
| 2 | **Traffic Class** (8-bits): These 8 bits are divided into two parts. The most significant 6 bits are used for Type of Service to let the Router Known what services should be provided to this packet. The least significant 2 bits are used for Explicit Congestion Notification (ECN). |
| 3 | **Flow Label** (20-bits): This label is used to maintain the sequential flow of the packets belonging to a communication. The source labels the sequence to help the router identify that a particular packet belongs to a specific flow of information. This field helps avoid re-ordering of data packets. It is designed for streaming/real-time media. |

4 **Payload Length** (16-bits): This field is used to tell the routers how much information a particular packet contains in its payload. Payload is composed of Extension Headers and Upper Layer data. With 16 bits, up to 65535 bytes can be indicated; but if the Extension Headers contain Hop-by-Hop Extension Header, then the payload may exceed 65535 bytes and this field is set to 0.

5 **Next Header** (8-bits): This field is used to indicate either the type of Extension Header, or if the Extension Header is not present then it indicates the Upper Layer PDU. The values for the type of Upper Layer PDU are same as IPv4's.

6 **Hop Limit** (8-bits): This field is used to stop packet to loop in the network infinitely. This is same as TTL in IPv4. The value of Hop Limit field is decremented by 1 as it passes a link (router/hop). When the field reaches 0 the packet is discarded.

7 **Source Address** (128-bits): This field indicates the address of originator of the packet.

8 **Destination Address** (128-bits): This field provides the address of intended recipient of the packet.

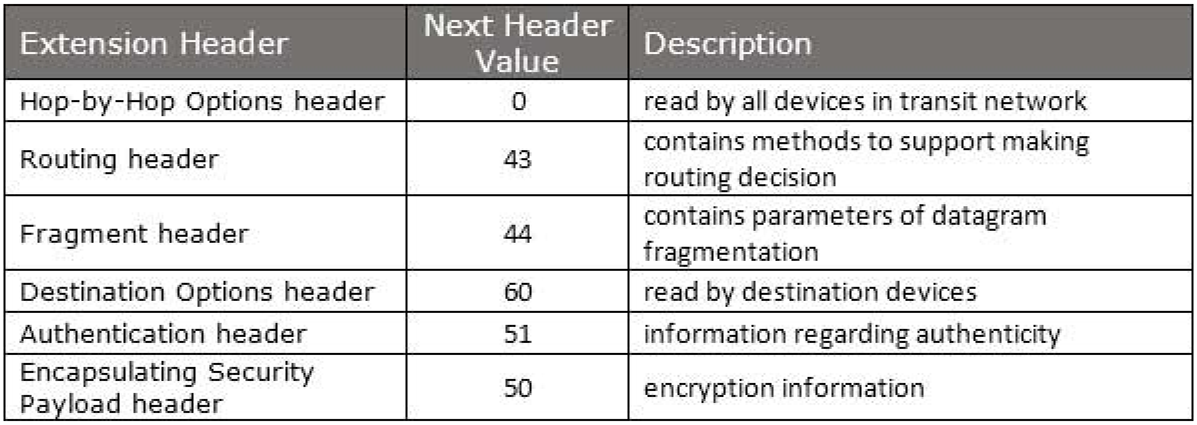
Extension Headers

In IPv6, the Fixed Header contains only that much information which is necessary, avoiding those information which is either not required or is rarely used. All such information is put between the Fixed Header and the Upper layer header in the form of Extension Headers. Each Extension Header is identified by a distinct value.

When Extension Headers are used, IPv6 Fixed Header's Next Header field points to the first Extension Header. If there is one more Extension Header, then the first Extension Header's 'Next-Header' field points to the second one, and so on. The last Extension Header's 'Next-Header' field points to the Upper Layer Header. Thus, all the headers points to the next one in a linked list manner.

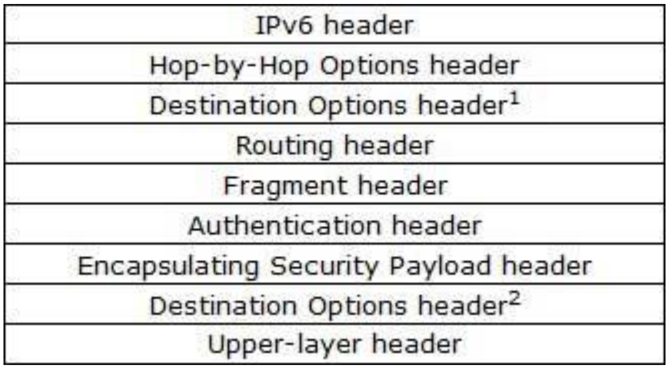
If the Next Header field contains the value 59, it indicates that there are no headers after this header, not even Upper Layer Header.

The following Extension Headers must be supported as per RFC 2460:



| Extension Header | Next Header Value | Description |
|---|---|---|
| Hop-by-Hop Options header | 0 | read by all devices in transit network |
| Routing header | 43 | contains methods to support making routing decision |
| Fragment header | 44 | contains parameters of datagram fragmentation |
| Destination Options header | 60 | read by destination devices |
| Authentication header | 51 | information regarding authenticity |
| Encapsulating Security Payload header | 50 | encryption information |

The sequence of Extension Headers should be:

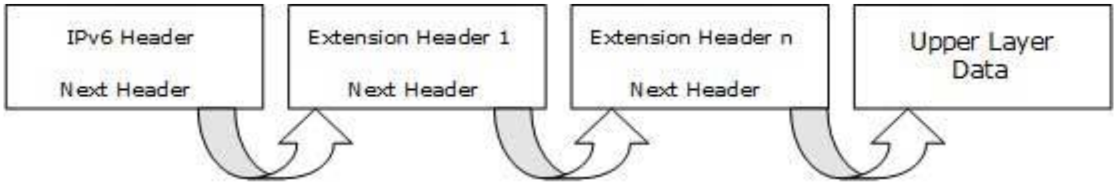


| IPv6 header |
|---|
| Hop-by-Hop Options header |
| Destination Options header[1] |
| Routing header |
| Fragment header |
| Authentication header |
| Encapsulating Security Payload header |
| Destination Options header[2] |
| Upper-layer header |

These headers:

- 1. should be processed by First and subsequent destinations.
- 2. should be processed by Final Destination.

Extension Headers are arranged one after another in a linked list manner, as depicted in the following diagram:



**37. Explain the IPV6 header format with neat sketch.**

[7M] **[Dec/Jan-2022-23] Understand CO2**

Ans) Internet Protocol Version 6 (IPv6) Header Format

IP version 6 is abbreviated as IPv6. It is an improved version of IPv4 in terms of efficiency and complexity and it is a newer version of Internet Protocol.

The **size** of the IPv6 address is **four** times greater than the size of the IPv4 address but the IPv6 **header size** is only **two** times greater than the IPv4 header size.

There is one **fixed size** header and zero or more than zero optional or extension headers in the IPv6 header. The fixed header keeps all the information that is important for the router. Optional information is kept in the extension header.

**List of IPv6 Header Format Components**

There are two main parts to the IPv6 data packet that is header and payload. The header of IPv6 is of a fixed length of 40 bytes which has the following fields:

Refer to the below image for the components of the IPv6 header

- Version
- Traffic Class
- Flow label:
- Payload Length (16-bits)
- Next Header (8-bits):
- Hop Limit (8-bits)
- Source Address (128 bits)
- Destination Address (128 bits)

**Version (4-bits)**

It shows the version of the internet protocol we used, i.e. 0110

**Traffic Class (8-bits)**

This is an 8-bit field in which 8 bits are divided into **two** parts. The most significant 6-bit is for the type of service so that the router will get to know about what services need to be provided to the given packet. And for **Explicit Congestion Notification** (ECN), the least significant 2-bit is used.

**Flow Label (20-bits)**

This 20-bit is required for maintaining the sequential flow of packets related to a particular communication. This field is also helpful in avoiding the reordering of packets. The source labels the sequence to help the router so that it can identify that a particular packet is related to a specific flow of data. It is generally used for real or streaming media.

**Payload Length (16-bits)**

This field is used to help the router know how much information is stored in the payload of a particular packet.

**Next Header (8-bits)**

This field is used to represent the type of extension header or if the extension header is not present then it shows the Upper Layer **PDU**. The value for Upper Layer PDU is the same as that of values in IPv4.

**Hop Limit (8-bits)**

Hop limit is a field in a header that stops the header from going into an infinite loop in the network. It works the same as that of **TTL** in IPv4. When it passes a hop or router its value is **decremented** by 1. The packet is discarded when it reaches 0.

**Source Address (128-bits)**

This field provides the address from where the packet originates.

**Destination Address (128-bits)**

The destination address is the address of the packet's intended recipient.

**38. Explain the transition from IPV4 to IPV6.**

[7M] **[Nov-2019] Understand CO2**

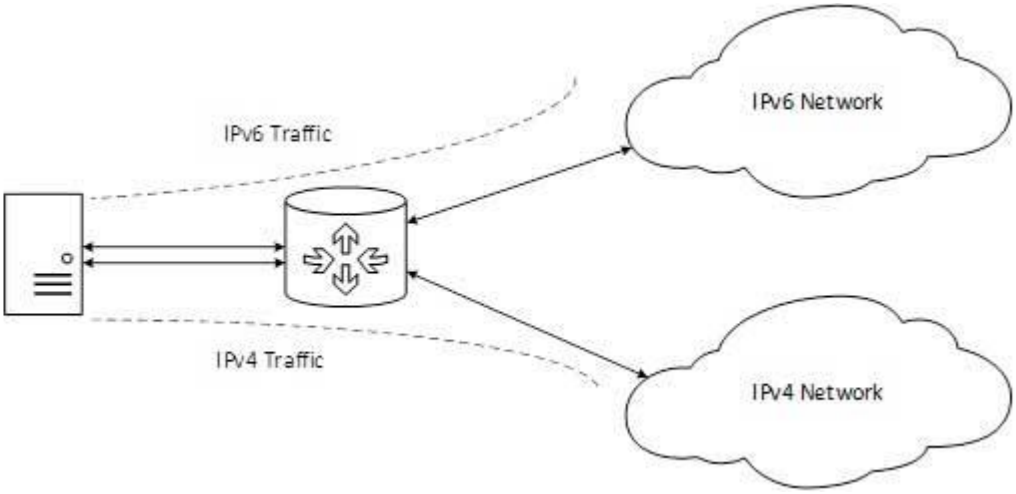
**Ans) Transition From IPv4 to IPv6**

Complete transition from IPv4 to IPv6 might not be possible because IPv6 is not backward compatible. This results in a situation where either a site is on IPv6 or it is not. It is unlike implementation of other new technologies where the newer one is backward compatible so the older system can still work with the newer version without any additional changes.

To overcome this short-coming, we have a few technologies that can be used to ensure slow and smooth transition from IPv4 to IPv6.

Dual Stack Routers

A router can be installed with both IPv4 and IPv6 addresses configured on its interfaces pointing to the network of relevant IP scheme.

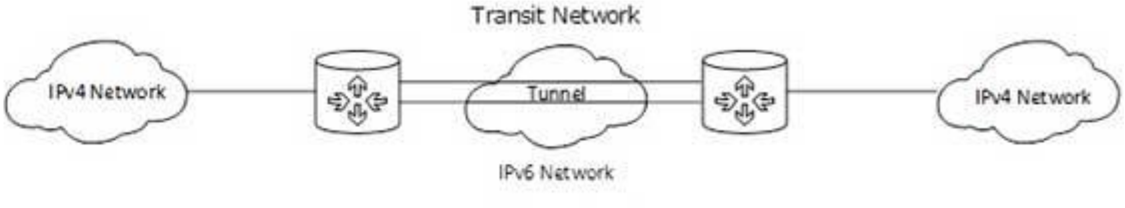


[Image: Dual Stack Router]

In the above diagram, a server having IPv4 as well as IPv6 address configured for it can now speak with all the hosts on both the IPv4 as well as the IPv6 networks with the help of a Dual Stack Router. The Dual Stack Router, can communicate with both the networks. It provides a medium for the hosts to access a server without changing their respective IP versions.

Tunneling

In a scenario where different IP versions exist on intermediate path or transit networks, tunneling provides a better solution where user's data can pass through a non-supported IP version.

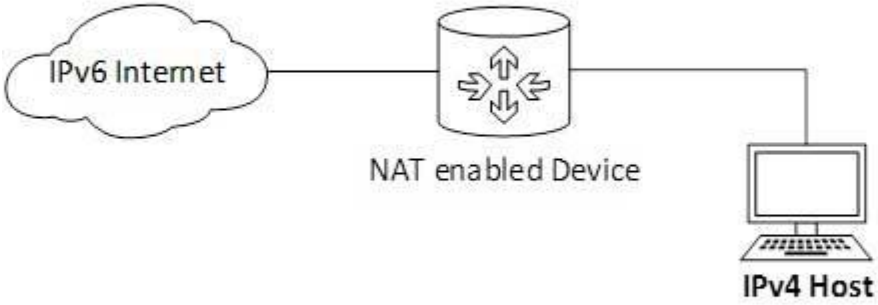


[Image: Tunneling]

The above diagram depicts how two remote IPv4 networks can communicate via a Tunnel, where the transit network was on IPv6. Vice versa is also possible where the transit network is on IPv6 and the remote sites that intend to communicate are on IPv4.

NAT Protocol Translation

This is another important method of transition to IPv6 by means of a NAT-PT (Network Address Translation – Protocol Translation) enabled device. With the help of a NAT-PT device, actual can take place happens between IPv4 and IPv6 packets and vice versa. See the diagram below:



[Image: NAT - Protocol Translation]

A host with IPv4 address sends a request to an IPv6 enabled server on Internet that does not understand IPv4 address. In this scenario, the NAT-PT device can help them communicate. When the IPv4 host sends a request packet to the IPv6 server, the NAT-PT device/router strips down the IPv4 packet, removes IPv4 header, and adds IPv6 header and passes it through the Internet. When a response from the IPv6 server comes for the IPv4 host, the router does vice versa.

**39. What is the major problem of IPV4 protocol? What are the solutions?**

**[7M][Supply-Jun/July-2022] Analysis CO2**

The major problem with the IPv4 protocol, which is the fourth version of the Internet Protocol and the foundation of the internet's addressing system, is the exhaustion of available IP addresses. IPv4 uses a 32-bit addressing scheme, allowing for approximately 4.3 billion unique addresses. With the exponential growth of internet-connected devices, this pool of addresses became insufficient, leading to IPv4 address exhaustion.

To address this issue, the technology community introduced several solutions:

1. Transition to IPv6: IPv6 (Internet Protocol version 6) uses a 128-bit addressing scheme, providing an almost unlimited number of unique IP addresses (about 340 undecillion).

Transitioning to IPv6 is a long-term solution to address the scarcity of IP addresses. It offers vast address space and enhances the security and efficiency of internet traffic.

2. Network Address Translation (NAT): NAT allows multiple devices within a private network to share a single public IP address. It conserves public IPv4 addresses by mapping multiple private addresses to a single public address. While NAT provides a short-term solution, it's not a scalable or ideal approach in the long run.

3. Private Addressing: Utilizing private IP address ranges within local networks, like the RFC 1918 addresses (e.g., 192.168.0.0 - 192.168.255.255), allows multiple devices to share a single public IP address. This is often used in combination with NAT.

4. IPv4 Address Sharing Technologies: Technologies like Carrier-Grade NAT (CGN or CGNAT) enable service providers to share a single public IPv4 address among multiple users. While this solution allows for the conservation of public addresses, it can complicate end-to-end connectivity.

5. Dual-Stack Implementation: This involves running both IPv4 and IPv6 simultaneously. It allows devices to communicate using either protocol, facilitating a smooth transition to IPv6 while ensuring compatibility with IPv4.

In the long term, the adoption of IPv6 is essential to overcome the limitations of IPv4. However, due to the extensive deployment and infrastructure changes required, the transition to IPv6 has been gradual. Nonetheless, it remains the most effective solution to address the limitation of IPv4 addresses.

**40. Explain the format of IPv4, IPv6 header, describe the significance of each field and the major problems of IPv4 protocol, and discuss the various classes of IPV4 addressing.**

Ans)IPv4 Header Format:

The IPv4 header is the initial part of an IP packet and contains essential information for routing and delivery.

1. Version (4 bits): Indicates the IP version. For IPv4, the value is 4.

2. Header Length (4 bits): Specifies the length of the IP header. It is needed as the header might contain additional options.

3. Type of Service (8 bits): Originally intended for QoS (Quality of Service) parameters, but not widely used.

4. Total Length (16 bits): Represents the entire packet size, including the header and payload.

5. Identification (16 bits): Used for uniquely identifying fragments of an IP datagram.

6. Flags (3 bits) and Fragment Offset (13 bits): Used for fragmentation and reassembly of IP datagrams.

7. Time to Live (TTL) (8 bits): Indicates the maximum number of hops a packet can take before being discarded to prevent routing loops.

8. Protocol (8 bits): Specifies the protocol used in the data portion of the packet (e.g., TCP, UDP).

9. Header Checksum (16 bits): Ensures the integrity of the header.

10. Source IP Address (32 bits) and Destination IP Address (32 bits):The source and destination addresses of the packet.

IPv6 Header Format:

The IPv6 header is simpler than IPv4, designed to improve routing and security.

1. Version (4 bits): Specifies the IP version. For IPv6, the value is 6.

2. Traffic Class (8 bits): Similar to the Type of Service in IPv4, meant for QoS. It prioritizes packets within the network.

3. Flow Label (20 bits):Used to identify and distinguish different classes of traffic for special handling by routers.

4. Payload Length (16 bits):Indicates the size of the payload (data), excluding the header.

5. Next Header (8 bits): Specifies the type of the next header (like the Protocol field in IPv4).

6. Hop Limit (8 bits): Similar to the Time to Live in IPv4, but it measures the number of hops instead of time.

7. Source IP Address (128 bits) and Destination IP Address (128 bits): Identifies the source and destination of the packet.

Major Problems with IPv4 Protocol:

The primary problem with IPv4 is the exhaustion of available IP addresses due to the 32-bit address space, limiting the number of unique addresses to about 4.3 billion. The growth in internet-connected devices far exceeds this capacity.

Classes of IPv4 Addressing:

IPv4 addresses are categorized into five classes: A, B, C, D, and E.

1. Class A (1.0.0.0 to 126.0.0.0): Supports a large number of networks but with a limited number of hosts per network.

2. Class B (128.0.0.0 to 191.255.0.0):Allows for a moderate number of networks and hosts.

3. Class C (192.0.0.0 to 223.255.255.0): Supports a large number of hosts but with fewer networks.

4. Class D (224.0.0.0 to 239.255.255.255): Reserved for multicast groups.

5. Class E (240.0.0.0 to 255.255.255.255): Reserved for experimental purposes and is not for public use.

These classes had fixed bit allocations for network and host portions, but due to the limitations of these fixed sizes and the depletion of available addresses, Classless Inter-Domain Routing (CIDR) was introduced to allocate addresses more efficiently.IPv4's limitations have led to the necessity of IPv6, which provides a much larger address space and several additional features to improve upon the shortcomings of IPv4.

**45. Explain about Class full addressing and CIDR.**

**Ans; Classful Address**

The first addressing system to be implemented as part of the Internet Protocol was Classful Addressing. In the year 1981, the Classful addressing network architecture was first used on the Internet. The Classful addressing system was superseded by a Classless addressing scheme with the introduction of Classless Inter-Domain Routing (CIDR) in 1993.

The IP address comprises up of 32 bits and is split into four sections separated by dots: **part 1, part 2, part 3, and part 4**.

The IP address is made up of four parts, each of which is eight bits long (1 byte).

Further, the 4 parts of the IP address is divided into parts: a **network ID** and a **Host ID**.

Types of Classful Address

In Class A, B, and C, the address **Class A, Class B, Class C, Class D, and Class E** are the five varieties of Classful addresses. In IPv4, this classification is known as Classful addressing or IP address classes.

The first three classes, Class A, B, and C, are used for "public addressing", in which communication is always one-to-one between source and destination. It implies that when data is transmitted from a source, it will only be sent to a single network host.

The reserved categories include Class D and Class E, with Class D being utilized for multicast and Class E being saved for future usage exclusively.

In IPv4, the Network ID is the first part of Class A, B, and C, while the Host ID is the remaining second portion.

The Host ID always indicates the number of hosts or nodes in a certain network, whereas the Network ID always space is split into a certain number of IP address blocks. It also specifies the maximum number of hosts in a network.

Network and Host part in Classful Addressing

The first octet or byte of an IP address is part of the network ID (short for Net-ID), while the next three octets or three bytes are part of the host ID in Class A. (in short, host-ID).

The network ID takes up the first two octets or two bytes in Class B, whereas the host ID takes up the remaining two octets or two bytes.

In Class C, the first three octets or bytes are dedicated to the network ID, while the last octet or byte is dedicated to the host ID.

## CIDR / Classless

With Classless Inter-Domain Routing (CIDR), IP assignments are not limited to the three classes. The whole unicast range (any IP address with a first octet of 0 – 223) can be allocated in any size block. In effect, the whole concept of IP address “classes” is done away with entirely.

Instead of requiring the IP assignment from the RIRs to be either a 255.0.0.0 or 255.255.0.0 or 255.255.255.0 block, they could be any size — and for simplicity, slash notation was adopted.

If you need 300 IP addresses … You get a /23.

If you need 500 IP addresses … You also get a /23.

If you need 1000 IP addresses … You get a /22.

If you need 25,000 IP addresses … You get a /17.

If you need 70,000 IP addresses … You get a /15.

If you need 250,000 IP addresses … You get a /14 (instead of the ~16 million IP addresses from the /8 block that would have been assigned in the Classful world).

This creates a system in which IP address ranges are assigned with a much, much smaller rate of wasted IP addresses.

CIDR address assignment was ratified in RFC 1518, back in September of 1993. Making it the ubiquitous standard for the last 30 years (if you’re reading this in 2023).

The concept of Classful address assignment is useful to know from a historical perspective. But in reality, nowhere in the world is Classful addressing still employed.

The rare exception, however, is certain archaic protocols or devices which operate “classfully”. This means they assume a mask based upon the IP address, according to the IP address’s class (i.e., an IP address’s first octet).

For example, if a classful protocol or device is given the IP address 199.22.33.4 — the first octet is 199, which means this is a Class C address, and the Subnet Mask is assumed to be 255.255.255.0.

**42. A router has received new IP addresses: 57.6.96.0/21, 57.6.104.0/21, 57.6.112.0/21 and 57.6.120.0/21. If all of them use the same outgoing line, can they be aggregated?**

**[7M] [Supply-June/July-2022] Analysis CO2**

**Ans:**

Yes, these IP address ranges can be aggregated because they are contiguous and have the same prefix length (/21). To aggregate them, you can find the smallest supernet that encompasses all of these ranges.

Here's how you can do it step by step:

1. Convert the IP addresses to binary format:

   - 57.6.96.0/21: 00111001.00000110.01100000.0000xxxx

   - 57.6.104.0/21: 00111001.00000110.01101000.0000xxxx

   - 57.6.112.0/21: 00111001.00000110.01110000.0000xxxx

   - 57.6.120.0/21: 00111001.00000110.01111000.0000xxxx

2. Identify the common prefix in binary form: - The common prefix for all four ranges is: 00111001.00000110.01xxxxxx.xxxxxxxx

3. Count the number of common bits in the prefix, which is 19 bits (the first 19 bits are the same for all four ranges).

4. Write the aggregated IP range:

   - 57.6.96.0/19

So, you can aggregate these four IP ranges into a single range: 57.6.96.0/19. This single range covers all the addresses in the original ranges.

**43. Illustrate subnetting in networks.**

**[7M] [Supply-June/July-2022] Analysis CO2**

**Ans:**

Subnetting is a technique used in networking to divide a larger IP network into smaller, more manageable subnetworks, or subnets. Subnetting is essential for optimizing network resources, improving security, and efficient IP address allocation. Here, I'll illustrate subnetting with an example.

Let's say you have been allocated the IP address range 192.168.1.0/24 (a common private IP address range) for your organization. The "/24" means that you have 256 addresses (from 192.168.1.0 to 192.168.1.255) in a single network.

Now, you want to subnet this network to create smaller, separate subnetworks. Here's how you can do it:

1. **Choose the Subnet Mask**: You need to decide on the new subnet mask based on how many subnets and hosts per subnet you need. Let's say you want four subnets and each subnet should support up to 30 hosts. To accommodate this, you would need at least 32 addresses per subnet (2^5 = 32), so you'll use a subnet mask of /27 (32 addresses, 5 bits for host addresses).

2. **Divide the Address Space**: With a /27 subnet mask, each subnet will have 32 addresses. You can divide the original /24 address space into smaller subnets like this:

- Subnet 1: 192.168.1.0/27 (Addresses: 192.168.1.1 to 192.168.1.30)

- Subnet 2: 192.168.1.32/27 (Addresses: 192.168.1.33 to 192.168.1.62)

- Subnet 3: 192.168.1.64/27 (Addresses: 192.168.1.65 to 192.168.1.94)

- Subnet 4: 192.168.1.96/27 (Addresses: 192.168.1.97 to 192.168.1.126)

3. **Assign IP Addresses**: Now, you can assign these subnets to different parts of your network. For example, you might allocate Subnet 1 to your HR department, Subnet 2 to the Sales department, Subnet 3 to the IT department, and Subnet 4 to the Finance department.

This subnetting process allows you to efficiently use your IP address space and isolate different parts of your network. It also enhances security by creating natural boundaries between different segments of your network.

Remember that when you subnet, you should account for the network and broadcast addresses within each subnet, which is why Subnet 1 starts at 192.168.1.1 and ends at 192.168.1.30, for example. The first and last addresses in each subnet are typically reserved for the network and broadcast addresses**.**

**44. Explain about class full addressing and CIDR.**

[7M] **[Dec/Jan-2022-23] Understand CO2**

**Ans:**

Classful addressing and CIDR (Classless Inter-Domain Routing) are two different approaches to IP address allocation and routing in computer networks. **Let'explore each of them:**

**Classful Addressing:**

In the early days of the Internet, IP address allocation followed a classful addressing scheme. Classful addressing divided IP addresses into three primary classes: Class A, Class B, and Class C. Each class had a fixed network portion and host portion.

1. **Class A Addresses:** Class A addresses allocated the first byte (8 bits) for network identification and left the remaining three bytes (24 bits) for host addresses. This allowed for a large number of hosts but a limited number of networks. Example: 10.0.0.0 to 10.255.255.255.

2. **Class B Addresses:** Class B addresses allocated the first two bytes (16 bits) for network identification and the remaining two bytes (16 bits) for hosts. This allowed for more networks but fewer hosts per network. Example: 172.16.0.0 to 172.31.255.255.

3. **Class C Addresses:** Class C addresses allocated the first three bytes (24 bits) for network identification and only one byte (8 bits) for hosts. This allowed for many networks but very few hosts per network. Example: 192.0.0.0 to 223.255.255.255.

Classful addressing had limitations and inefficiencies. It didn't easily accommodate the varying needs of different organizations and led to IP address wastage.

**CIDR (Classless Inter-Domain Routing):**

CIDR was introduced to address the limitations of classful addressing and provide more flexibility in IP address allocation and routing. With CIDR:

1. **Variable-Length Subnet Masking (VLSM):** CIDR allows the allocation of address blocks with variable lengths of subnet masks. This means you can allocate subnets of different sizes to suit specific network requirements. You are no longer restricted to fixed class boundaries.

2. **Aggregation:** CIDR enables the aggregation of multiple IP address blocks into a single, summarized address, reducing the size of routing tables. This helps improve the efficiency of routing on the Internet.

3. **Classless:** CIDR is "classless" because it doesn't strictly follow the Class A, B, or C boundaries. It allows the allocation of IP addresses based on a prefix length (e.g., /24, /25) rather than predefined classes.

CIDR notation represents an IP address and its associated prefix length, denoted as "IP_address/prefix_length." For example, 192.168.1.0/24 represents a Classless Inter-Domain Routing (CIDR) notation indicating that the first 24 bits are the network portion, allowing for 256 host addresses.

CIDR has greatly improved the efficient utilization of IP address space and routing on the Internet by enabling finer-grained control over address allocation and reducing the size of routing tables. It has become the standard for IP address allocation and routing.

**45. Perform CIDR aggregation on the following IP addresses: 128.56.24.0/24, 128.56.25.0/24, 128.56.27.0/24 (OR) 128.56.26.0/24 [7M][Supply July – 2023] Analyze CO**

**Ans:**

**To perform CIDR aggregation on the given IP addresses, you'll need to find the common prefix among them and then express that common prefix using CIDR notation. Let's do it for the provided IP addresses: 128.56.24.0/24, 128.56.25.0/24, 128.56.27.0/24, and 128.56.26.0/24.**

1. Convert the IP addresses to binary form and compare them to find the common prefix:

- 128.56.24.0/24: 10000000.00111000.00011000.0000xxxx

- 128.56.25.0/24: 10000000.00111000.00011001.0000xxxx

- 128.56.27.0/24: 10000000.00111000.00011011.0000xxxx

- 128.56.26.0/24: 10000000.00111000.00011010.0000xxxx

2. Identify the common prefix:

- The common prefix is 128.56.24.0/22 because the first 22 bits are the same for all four IP addresses.

3. Write the aggregated IP address using CIDR notation:  - 128.56.24.0/22

So, the CIDR-aggregated notation for the given IP addresses 128.56.24.0/24, 128.56.25.0/24, 128.56.27.0/24, and 128.56.26.0/24 is 128.56.24.0/22. This aggregated range covers all the addresses in the original ranges while providing a more concise representation.