Advanced Computer Networks

Unit 1

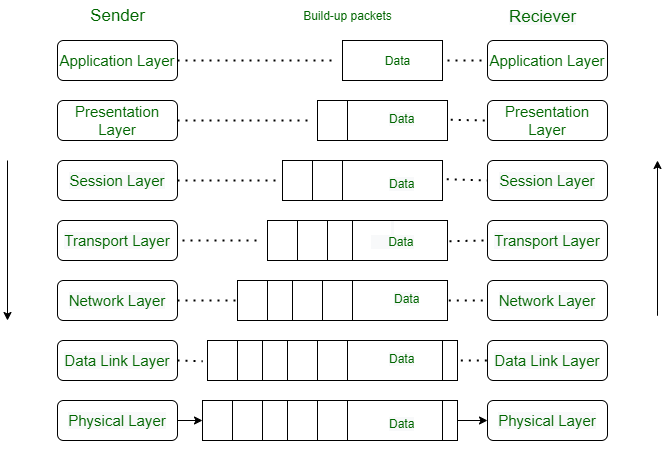
**Computer Network**

A computer network is a collection of interconnected devices, such as computers, servers, routers, switches, and other hardware components, that are linked together to facilitate communication and data exchange. These networks can be established using various technologies and protocols, enabling devices to share resources, exchange information, and collaborate on tasks.

1. **Virtual private network (VPN)**: VPN is an overlay private network stretched on top of a public network.
2. **Cloud network**: Technically, a cloud network is a WAN whose infrastructure is delivered via cloud services.
3. **Personal area network (PAN)**: PAN refers to a network used by just one person to connect multiple devices, such as laptops to scanners, etc.
4. **Local area network (LAN)**: The local area network connects devices within a limited geographical area, such as schools, hospitals, or office buildings.
5. **Metropolitan area network (MAN)**: MAN is a large computer network that spans across a city.
6. **Wide area network (WAN)**: Wide area networks cover larger areas such as large cities, states, and even countries.

**OSI Model**

The OSI model, created in 1984 by ISO, is a reference framework that explains the process of transmitting data between computers. It is divided into seven layers that work together to carry out specialised network functions, allowing for a more systematic approach to networking.



**Physical Layer – Layer 1**

The lowest layer of the OSI reference model is the physical layer. It is responsible for the actual physical connection between the devices. The physical layer contains information in the form of**bits.** It is responsible for transmitting individual bits from one node to the next. When receiving data, this layer will get the signal received and convert it into 0s and 1s and send them to the Data Link layer, which will put the frame back together.

**Functions of the Physical Layer**

* **Bit synchronization:** The physical layer provides the synchronization of the bits by providing a clock. This clock controls both sender and receiver thus providing synchronization at the bit level.
* **Bit rate control:** The Physical layer also defines the transmission rate i.e. the number of bits sent per second.
* **Physical topologies:** Physical layer specifies how the different, devices/nodes are arranged in a network i.e. bus, star, or mesh topology.
* **Transmission mode:** Physical layer also defines how the data flows between the two connected devices. The various transmission modes possible are Simplex, half-duplex and full-duplex.

**Data Link Layer (DLL) – Layer 2**

The data link layer is responsible for the node-to-node delivery of the message. The main function of this layer is to make sure data transfer is error-free from one node to another, over the physical layer. When a packet arrives in a network, it is the responsibility of the DLL to transmit it to the Host using its MAC address. The Data Link Layer is divided into two sublayers:

1. Logical Link Control (LLC)
2. Media Access Control (MAC)

The packet received from the Network layer is further divided into frames depending on the frame size of the NIC(Network Interface Card). DLL also encapsulates Sender and Receiver’s MAC address in the header.

The Receiver’s MAC address is obtained by placing an ARP(Address Resolution Protocol) request onto the wire asking “Who has that IP address?” and the destination host will reply with its MAC address.

**Functions of the Data Link Layer**

* **Framing:**Framing is a function of the data link layer. It provides a way for a sender to transmit a set of bits that are meaningful to the receiver. This can be accomplished by attaching special bit patterns to the beginning and end of the frame.
* **Physical addressing:** After creating frames, the Data link layer adds physical addresses (MAC addresses) of the sender and/or receiver in the header of each frame.
* **Error control:** The data link layer provides the mechanism of error control in which it detects and retransmits damaged or lost frames.
* **Flow Control:** The data rate must be constant on both sides else the data may get corrupted thus, flow control coordinates the amount of data that can be sent before receiving an acknowledgment.
* **Access control:**When a single communication channel is shared by multiple devices, the MAC sub-layer of the data link layer helps to determine which device has control over the channel at a given time.

**Network Layer – Layer 3**

The network layer works for the transmission of data from one host to the other located in different networks. It also takes care of packet routing i.e. selection of the shortest path to transmit the packet, from the number of routes available. The sender & receiver’s IP addresses are placed in the header by the network layer.

**Functions of the Network Layer**

* **Routing:** The network layer protocols determine which route is suitable from source to destination. This function of the network layer is known as routing.
* **Logical Addressing:**To identify each device on Internetwork uniquely, the network layer defines an addressing scheme. The sender & receiver’s IP addresses are placed in the header by the network layer. Such an address distinguishes each device uniquely and universally.

**Transport Layer – Layer 4**

The transport layer provides services to the application layer and takes services from the network layer. The data in the transport layer is referred to as *Segments*. It is responsible for the End to End Delivery of the complete message. The transport layer also provides the acknowledgment of the successful data transmission and re-transmits the data if an error is found.

**At the sender’s side:**The transport layer receives the formatted data from the upper layers, performs **Segmentation**, and also implements **Flow & Error control** to ensure proper data transmission. It also adds Source and Destination port numbers in its header and forwards the segmented data to the Network Layer.

**Functions of the Transport Layer**

* **Segmentation and Reassembly:** This layer accepts the message from the (session) layer, and breaks the message into smaller units. Each of the segments produced has a header associated with it. The transport layer at the destination station reassembles the message.
* **Service Point Addressing:** To deliver the message to the correct process, the transport layer header includes a type of address called service point address or port address. Thus by specifying this address, the transport layer makes sure that the message is delivered to the correct process.
* **Services Provided by Transport Layer**
  1. **Connection-Oriented Service:** It is a three-phase process that includes:
* Connection Establishment
* Data Transfer
* Termination/disconnection

In this type of transmission, the receiving device sends an acknowledgment, back to the source after a packet or group of packets is received. This type of transmission is reliable and secure.

**2. Connectionless service:** It is a one-phase process and includes Data Transfer. In this type of transmission, the receiver does not acknowledge receipt of a packet. This approach allows for much faster communication between devices. Connection-oriented service is more reliable than connectionless Service.

**Session Layer – Layer 5**

This layer is responsible for the establishment of connection, maintenance of sessions, and authentication, and also ensures security.

**Functions of the Session Layer**

* **Session establishment, maintenance, and termination:** The layer allows the two processes to establish, use and terminate a connection.
* **Synchronization:** This layer allows a process to add checkpoints that are considered synchronization points in the data. These synchronization points help to identify the error so that the data is re-synchronized properly, and ends of the messages are not cut prematurely and data loss is avoided.
* **Dialog Controller:** The session layer allows two systems to start communication with each other in half-duplex or full-duplex.

**Presentation Layer – Layer 6**

The presentation layer is also called the **Translation layer**. The data from the application layer is extracted here and manipulated as per the required format to transmit over the network.

**Functions of the Presentation Layer**

* **Encryption/ Decryption:** Data encryption translates the data into another form or code. The encrypted data is known as the ciphertext and the decrypted data is known as plain text. A key value is used for encrypting as well as decrypting data.
* **Compression:** Reduces the number of bits that need to be transmitted on the network.

**Application Layer – Layer 7**

At the very top of the OSI Reference Model stack of layers, we find the Application layer which is implemented by the network applications. These applications produce the data, which has to be transferred over the network. This layer also serves as a window for the application services to access the network and for displaying the received information to the user.

**Functions of the Application Layer**

The main functions of application layer are given below.

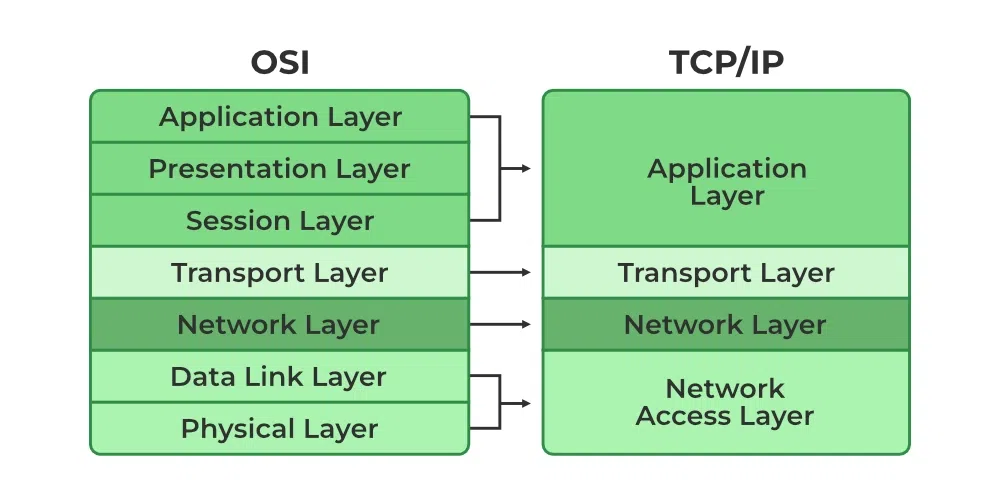
* Network Virtual Terminal: It allows a user to log on to a remote host.
* FTAM- File transfer access and management: This application allows a user to  
  access file in a remote host, retrieve files in remote host and manage or  
  control files from a remote computer.
* Mail Services : Provide email service.
* Directory Services : This application provides distributed database sources  
  and access for global information about various objects and services.

**TCP/IP**

**TCP/IP** was designed and developed by the Department of Defense (DoD) in the 1960s and is based on standard protocols. It stands for Transmission Control Protocol/Internet Protocol. The TCP/IP model is a concise version of the OSI model. It contains four layers, unlike the seven layers in the OSI model.

The main work of TCP/IP is to transfer the data of a computer from one device to another. The main condition of this process is to make data reliable and accurate so that the receiver will receive the same information which is sent by the sender. To ensure that, each message reaches its final destination accurately, the TCP/IP model divides its data into packets and combines them at the other end, which helps in maintaining the accuracy of the data while transferring from one end to another end.

TCP and IP are different protocols of Computer Networks. The basic difference between TCP (Transmission Control Protocol) and IP (Internet Protocol) is in the transmission of data. In simple words, IP finds the destination of the mail and TCP has the work to send and receive the mail. UDP is another protocol, which does not require IP to communicate with another computer. IP is required by only TCP. This is the basic difference between TCP and IP.

****

**Network Access Layer**

* A network layer is the lowest layer of the TCP/IP model.
* A network layer is the combination of the Physical layer and Data Link layer defined in the OSI reference model.
* It defines how the data should be sent physically through the network.
* This layer is mainly responsible for the transmission of the data between two devices on the same network.
* The functions carried out by this layer are encapsulating the IP datagram into frames transmitted by the network and mapping of IP addresses into physical addresses.
* The protocols used by this layer are ethernet, token ring, FDDI, X.25, frame relay.

**Internet Layer**

* An internet layer is the second layer of the TCP/IP model.
* An internet layer is also known as the network layer.
* The main responsibility of the internet layer is to send the packets from any network, and they arrive at the destination irrespective of the route they take.

**Following are the protocols used in this layer are:**

* **IP Protocol:** IP protocol is used in this layer, and it is the most significant part of the entire TCP/IP suite.
* **ARP Protocol**: ARP stands for Address Resolution Protocol. ARP is a network layer protocol which is used to find the physical address from the IP address.
* **ICMP Protocol**: ICMP stands for Internet Control Message Protocol. It is a mechanism used by the hosts or routers to send notifications regarding datagram problems back to the sender. A datagram travels from router-to-router until it reaches its destination. If a router is unable to route the data because of some unusual conditions such as disabled links, a device is on fire or network congestion, then the ICMP protocol is used to inform the sender that the datagram is undeliverable.

**Transport Layer**

The transport layer is responsible for the reliability, flow control, and correction of data which is being sent over the network.

The two protocols used in the transport layer are User Datagram protocol and Transmission control protocol.

* **User Datagram Protocol (UDP):** It provides connectionless service and end-to-end delivery of transmission. User Datagram Protocol discovers the error, and ICMP protocol reports the error to the sender that user datagram has been damaged.
* **Transmission Control Protocol (TCP):** It provides a full transport layer services to applications.It creates a virtual circuit between the sender and receiver, and it is active for the duration of the transmission. TCP divides the whole message into smaller units known as segment, and each segment contains a sequence number which is required for reordering the frames to form an original message.

**Application Layer**

* An application layer is the topmost layer in the TCP/IP model.
* It is responsible for handling high-level protocols, issues of representation.
* This layer allows the user to interact with the application.
* When one application layer protocol wants to communicate with another application layer, it forwards its data to the transport layer.
* There is an ambiguity occurs in the application layer. Every application cannot be placed inside the application layer except those who interact with the communication system. For example: text editor cannot be considered in application layer while web browser using HTTP protocol to interact with the network where HTTP protocol is an application layer protocol.

Following are the main protocols used in the application layer:

* **HTTP:** HTTP stands for Hypertext transfer protocol. This protocol allows us to access the data over the world wide web. It transfers the data in the form of plain text, audio, video. It is known as a Hypertext transfer protocol as it has the efficiency to use in a hypertext environment where there are rapid jumps from one document to another.
* **SNMP:** SNMP stands for Simple Network Management Protocol. It is a framework used for managing the devices on the internet by using the TCP/IP protocol suite.
* **SMTP**: SMTP stands for Simple mail transfer protocol. The TCP/IP protocol that supports the e-mail is known as a Simple mail transfer protocol. This protocol is used to send the data to another e-mail address.
* **DNS**: DNS stands for Domain Name System. An IP address is used to identify the connection of a host to the internet uniquely. But, people prefer to use the names instead of addresses. Therefore, the system that maps the name to the address is known as Domain Name System.
* **FTP**: FTP stands for File Transfer Protocol. FTP is a standard internet protocol used for transmitting the files from one computer to another computer.

**MAC protocols for high speed LAN MANS Wireless LANS**

For high-speed LANs (Local Area Networks), MANs (Metropolitan Area Networks), and wireless LANs (Local Area Networks), several MAC (Medium Access Control) protocols have been developed to efficiently manage the access to the shared communication medium. Here are some commonly used MAC protocols for these network types:

1. **Ethernet (IEEE 802.3):**
   * Ethernet is one of the most prevalent LAN technologies. It uses Carrier Sense Multiple Access with Collision Detection (CSMA/CD) as its MAC protocol. However, for high-speed LANs, especially with the advent of gigabit and 10-gigabit Ethernet, CSMA/CD is largely replaced by full-duplex Ethernet, where collisions are avoided by using switches instead of hubs.
2. **Token Ring (IEEE 802.5):**
   * Token Ring is another LAN protocol where stations are arranged in a ring topology, and a token is passed around to control access to the shared medium. This protocol is less common compared to Ethernet but was historically used in some high-speed LAN environments.
3. **Fast Ethernet (IEEE 802.3u):**
   * Fast Ethernet extends the Ethernet standard to provide higher data rates, typically 100 Mbps. It still uses CSMA/CD but operates at higher speeds.
4. **Gigabit Ethernet (IEEE 802.3z/ab):**
   * Gigabit Ethernet further extends Ethernet to provide data rates up to 1 Gbps (1000 Mbps) or more. It typically operates in full-duplex mode, utilizing switches to eliminate collisions.
5. **10-Gigabit Ethernet (IEEE 802.3ae):**
   * This Ethernet standard supports data rates of 10 Gbps, again typically operating in full-duplex mode.

For MANs and wireless LANs, different MAC protocols are often used due to the different nature of these networks:

1. **WiMAX (IEEE 802.16):**
   * WiMAX is a wireless MAN technology that offers high-speed broadband connectivity over a wide area. It uses a variety of MAC protocols depending on the specific version and configuration, including variants of Time Division Multiple Access (TDMA) and Frequency Division Multiple Access (FDMA).
2. **Wi-Fi (IEEE 802.11):**
   * Wi-Fi is a popular wireless LAN technology used for local area networking. It uses CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) as its MAC protocol. Variants of Wi-Fi, such as 802.11ac and 802.11ax, provide high-speed connectivity in the 5 GHz frequency band and offer enhancements for improved performance in high-density environments.
3. **LTE (Long-Term Evolution):**
   * LTE is a wireless communication standard for high-speed data transmission over cellular networks, typically used for wide-area coverage. It employs orthogonal frequency-division multiple access (OFDMA) and SC-FDMA (Single Carrier Frequency Division Multiple Access) as its MAC protocols to manage multiple users efficiently.

**Fast access technologies**

Fast access technologies in computer networks refer to methods and technologies that improve the speed and efficiency of accessing data and resources over a network. These technologies are crucial for enhancing network performance, reducing latency, and improving overall user experience. Here are some common fast access technologies used in computer networks:

1. **Caching**: Caching involves storing frequently accessed data closer to the user, either on the client-side or on intermediary servers (such as proxy servers or Content Delivery Networks - CDNs). This reduces the need to retrieve data from the original source every time it's requested, resulting in faster access times.
2. **Content Delivery Networks (CDNs):** CDNs are a network of distributed servers strategically placed across different geographical locations. They store cached copies of web content (such as images, videos, and web pages) closer to users, reducing latency and improving load times, especially for users accessing content from distant locations.
3. **Load Balancing**: Load balancing distributes incoming network traffic across multiple servers to optimize resource utilization, maximize throughput, and minimize response times. By evenly distributing workload among servers, load balancers prevent any single server from becoming overwhelmed, thus ensuring fast access to resources.
4. **Data Compression**: Data compression techniques reduce the size of data transmitted over the network, thereby reducing bandwidth usage and improving transfer speeds. Compression algorithms like gzip or Brotli are commonly used to compress text-based data (e.g., HTML, CSS, JavaScript), while media codecs (e.g., MP3, JPEG) are used for compressing multimedia content.
5. **HTTP/2 and HTTP/3**: These are newer versions of the Hypertext Transfer Protocol (HTTP) designed to improve web page loading times. HTTP/2 introduces features like multiplexing, header compression, and server push, while HTTP/3 uses the QUIC protocol to further optimize performance by reducing latency and improving security.
6. **Solid-State Drives (SSDs):** SSDs offer faster data access speeds compared to traditional hard disk drives (HDDs) due to their lack of moving parts and use of flash memory. Incorporating SSDs into network storage solutions can significantly reduce data retrieval times and improve overall system responsiveness.
7. **Anycast**: Anycast is a networking technique that routes traffic to the nearest (in terms of network distance) among multiple destinations that share the same IP address. This can be used to improve the speed of accessing content or services by directing users to the nearest server or data center.
8. **Content Preloading**: Content preloading anticipates user actions and pre-fetches or preloads resources that are likely to be requested in the future. By proactively fetching data before it's actually needed, this technique can reduce perceived latency and improve responsiveness, especially for interactive web applications.

Unit 2

**IPv6**

**IPv6 is the newest version of internet protocol formulated by the Internet Engineering Task Force (IETF), which helps identify and local endpoint systems on a computer network and route online traffic while addressing the problem of IPv4 address depletion due to prolonged internet use worldwide.**

Internet protocol version 6 (IPv6) is a network layer protocol that allows communication to take place over the network. Each device on the internet has a unique IP address used to identify it and figure out where it is. At the time of the digital revolution of the 1990s, it became apparent that the IP addresses that Internet Protocol version 4 (IPv4) used to connect devices would not be enough to meet demand.

## **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.

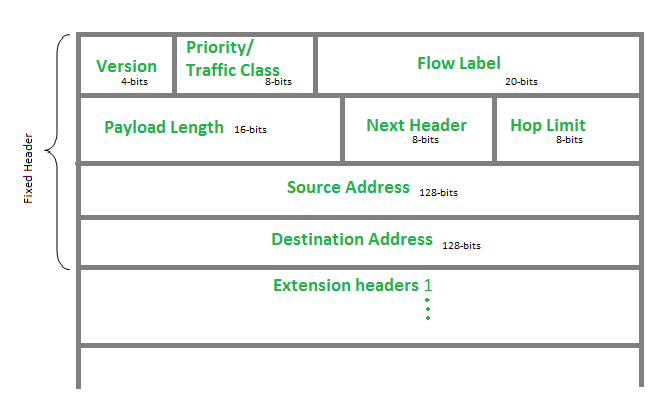
In IPv6 representation, we have three addressing methods :

* Unicast
* Multicast
* Anycast

## 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).

**IP version 6 Header Format :** 



**1. QoS challenges for IPv6**

One of the main challenges for QoS in IPv6 is the heterogeneity and diversity of IoT and smart city applications, devices, networks, and users. IoT and smart city applications can range from critical services such as health care, public safety, and traffic management, to non-critical services such as entertainment, education, and social media. Each of these applications may have different QoS requirements in terms of bandwidth, latency, jitter, packet loss, availability, and security. Moreover, IoT and smart city devices can vary in their capabilities, resources, power consumption, and mobility. Some devices may be constrained by battery life, memory, processing, or connectivity, while others may be more powerful and flexible. Similarly, IoT and smart city networks can involve different technologies, architectures, and protocols, such as wireless, wired, cellular, satellite, mesh, or cloud. Finally, IoT and smart city users can have different preferences, expectations, and priorities for accessing and using the applications and services.

* 1. **QoS mechanisms for IPv6**

To address the QoS challenges for IPv6, several mechanisms and techniques can be employed at different layers of the network stack. Traffic classification and marking, for example, involves labeling different types of traffic according to their QoS requirements and characteristics. Utilizing the 8-bit Differentiated Services Code Point (DSCP) field in the IPv6 header allows routers and switches to apply different policies and treatments to different traffic classes, such as prioritization, queuing, scheduling, or shaping. Traffic engineering and routing, on the other hand, optimizes the path and resources for traffic flows across the network based on their QoS requirements and network conditions. Utilizing the 20-bit Flow Label field in the IPv6 header allows traffic to be assigned to specific flows that are routed along predetermined paths with reserved bandwidth and resources. Additionally, resource reservation and admission control allocates and manages the network resources for traffic flows based on their QoS requirements and availability. Using protocols such as Resource Reservation Protocol (RSVP), or its extensions, traffic can request and reserve resources along the path from the source to the destination. Admission control is also employed to accept or reject traffic based on the network capacity and QoS policies.

* 1. **QoS standards for IPv6**

In addition to the QoS mechanisms for IPv6, there are also standards and frameworks that provide guidelines and recommendations for QoS in IPv6-based IoT and smart city applications. For instance, IEEE 802.1Q defines the Virtual Local Area Network (VLAN) technology which can be used to isolate and prioritize traffic. IEEE 802.11e defines the Enhanced Distributed Channel Access (EDCA) mechanism, a QoS enhancement for wireless networks based on the IEEE 802.11 family of protocols. IETF 6TiSCH is a framework that defines the Time-Slotted Channel Hopping (TSCH) mode of operation for low-power and lossy networks, which can be used to support various IoT and smart city applications that require low latency, high reliability, and low power consumption. Allowing for end-to-end QoS across different networks, these standards and frameworks can be integrated with DSCP values in the IPv6 header.

**IPv6 Security**

* IPv6 is the next generation of Internet protocol standard that is going to replace IPv4, the current standard of the Internet but is exhausting with its available address space.
* The IPv6 protocol has solved some, but not all, of the security problems found in IPv4 networks. One example is the mandatory inclusion of IP Security (IPsec) in the IPv6 protocol, which makes it fundamentally more secure than the older IPv4 standard. However, given its flexibility, the IPv6 protocol introduces new problems. A mobile IP protocol is built into the IPv6 protocol, and security solutions for this protocol are still under development.
* In addition, the dynamic configuration flexibility of IPv6 (such as stateless address auto-configuration) could also become a serious security problem, if not implemented correctly. The overall enhancements in IPv6 may provide better security in certain areas, but there are areas that attackers may be able to exploit.

**Neighbor Discovery**

IPv6 Neighbor Discovery (ND) is a protocol used in IPv6 networks to manage the interaction between neighboring devices on the same link (local network segment). It serves several functions that are similar to those performed by Address Resolution Protocol (ARP) in IPv4 networks but includes additional features and improvements. Here are the key functions and features of IPv6 Neighbor Discovery:

1. **Address Resolution**: Just like ARP in IPv4, Neighbor Discovery resolves IPv6 addresses to link-layer (MAC) addresses. This is accomplished through the Neighbor Solicitation (NS) and Neighbor Advertisement (NA) messages. When a node wants to communicate with another node on the same link, it first sends an NS message to resolve the link-layer address associated with the IPv6 address it wants to communicate with. The recipient responds with a NA message containing its link-layer address.
2. **Router Discovery**: IPv6 Neighbor Discovery enables hosts to discover routers on the local network segment. Routers periodically send Router Advertisement (RA) messages to announce their presence and provide network configuration parameters, such as prefix information and hop limits. Hosts use these messages to configure their IPv6 addresses and default gateway information.
3. **Address Autoconfiguration**: Neighbor Discovery simplifies the process of assigning IPv6 addresses to hosts by supporting stateless address autoconfiguration. Hosts can generate their IPv6 addresses based on prefixes provided in Router Advertisement messages. This eliminates the need for manual configuration or DHCP servers for address assignment in many cases.
4. **Duplicate Address Detection (DAD)**: Before assigning an IPv6 address to an interface, Neighbor Discovery performs duplicate address detection to ensure that the address is unique on the local link. This prevents address conflicts that could disrupt network communication.
5. **Neighbor Unreachability Detection**: Neighbor Discovery monitors the reachability of neighboring nodes by periodically sending Neighbor Solicitation messages. If a node does not receive a response to its solicitations, it considers the neighbor to be unreachable and takes appropriate action, such as updating its neighbor cache or triggering route changes.
6. **Router Redirection**: Routers can inform hosts about more efficient paths to specific destinations by sending Router Advertisement messages with redirection information. Hosts can then update their routing tables accordingly to improve packet forwarding efficiency.

**IPv6 autoconfiguration**

IPv6 autoconfiguration is a mechanism that enables IPv6 hosts to automatically configure their network interfaces with IPv6 addresses without manual intervention or the need for a DHCP (Dynamic Host Configuration Protocol) server. This process is designed to simplify IPv6 address assignment and make network setup more efficient. There are two main types of IPv6 autoconfiguration:

1. **Stateless Address Autoconfiguration (SLAAC)**: SLAAC is the primary method of IPv6 autoconfiguration and is specified in RFC 4862. With SLAAC, hosts generate their IPv6 addresses using information obtained from Router Advertisement (RA) messages sent by neighboring routers. The RA messages include network prefix information that hosts use to form their addresses, typically by combining the prefix with the interface's unique identifier (based on the interface's MAC address or another method like privacy extensions).

* The steps of SLAAC are as follows:
  + A host receives one or more Router Advertisement messages from neighboring routers.
  + The Router Advertisement messages contain the network prefix and other configuration parameters.
  + The host combines the network prefix with its interface identifier to form a unique IPv6 address.
  + Duplicate Address Detection (DAD) is performed to ensure the uniqueness of the address on the local link.
* SLAAC is efficient and lightweight, making it suitable for most IPv6 networks. It does not require the use of a DHCP server for address assignment, reducing configuration complexity.

1. **Stateful Address Autoconfiguration**: While SLAAC is the preferred method for IPv6 address assignment, stateful address autoconfiguration allows hosts to obtain additional configuration parameters, such as DNS server addresses or other custom settings, from a DHCPv6 server. Hosts can use DHCPv6 in conjunction with SLAAC if additional configuration beyond address assignment is needed.

* The steps of stateful address autoconfiguration involve:
  + Hosts receive Router Advertisement messages as in SLAAC.
  + In addition to SLAAC, hosts may use DHCPv6 to obtain additional configuration parameters if DHCPv6 servers are available on the network.

**IPv6 routing**

IPv6 routing refers to the process of forwarding IPv6 packets from a source node to a destination node across one or more interconnected networks or subnetworks. Routing in IPv6 networks is similar in concept to routing in IPv4 networks, but it operates within the context of the IPv6 protocol suite. Here's an overview of key aspects of IPv6 routing:

1. **Routing Tables**: Like IPv4, IPv6 routers maintain routing tables that contain information about the available paths to reach different IPv6 networks or destinations. These routing tables may be populated manually by network administrators or dynamically through routing protocols.
2. **Routing Protocols**: IPv6 routers use routing protocols to exchange routing information with neighboring routers and to dynamically update their routing tables. Common routing protocols used in IPv6 networks include:
   * **Routing Information Protocol version 6 (RIPng)**: A distance-vector routing protocol designed for IPv6.
   * **Open Shortest Path First version 3 (OSPFv3)**: A link-state routing protocol that supports IPv6.
   * **Border Gateway Protocol version 4 (BGP4+)**: Used for inter-domain routing between autonomous systems, with extensions to support IPv6 (BGP+).
   * **Intermediate System to Intermediate System version 6 (IS-ISv6)**: A link-state routing protocol similar to OSPFv3, commonly used in Service Provider networks.
3. **Router Advertisement and Router Solicitation**: IPv6 routers periodically send Router Advertisement (RA) messages to announce their presence and provide network configuration information, including prefixes and routing information. Hosts use Router Solicitation (RS) messages to request Router Advertisement messages when they join a network or when they need to refresh their network configuration.
4. **Prefix Delegation**: IPv6 routers may delegate prefixes to downstream routers or hosts, enabling hierarchical addressing and routing within the network. Prefix delegation is commonly used in broadband access networks (e.g., DHCPv6-PD) to assign prefixes to customer premises equipment (CPE) routers.
5. **IPv6 Addressing**: IPv6 routing operates based on IPv6 addresses, which are 128 bits in length and are typically assigned hierarchically based on network topology and addressing policies. IPv6 addresses may be assigned statically or dynamically through stateless or stateful address autoconfiguration.
6. **Routing Policies and Filters**: IPv6 routers can implement routing policies and filters to control the flow of IPv6 traffic based on various criteria, such as source or destination addresses, packet attributes, or routing metrics. This allows network administrators to enforce security policies, optimize traffic routing, and manage network resources.

**IPv6 Application Programming Interface**

The IPv6 API, also known as the IPv6 Application Programming Interface, provides a set of functions and protocols for developers to interact with and manage IPv6 networking features programmatically. It allows software applications to utilize IPv6 capabilities, such as addressing, routing, socket operations, and network configuration, within their codebase.

Here's an overview of some common components of the IPv6 API:

1. **Socket Programming APIs**: Many programming languages provide socket programming APIs that support IPv6, allowing developers to create networked applications that can communicate using IPv6. Examples include, **java.net** package (Java), **net** package (**Go**), **socket** module (**Python**) etc
2. **Network Configuration and Management APIs**: APIs and libraries for configuring and managing network interfaces and settings may include support for IPv6. For example:
   * **Netlink API**: Used in Linux for network configuration and monitoring,
   * **Windows Management Instrumentation (WMI)**: Provides access to network configuration information on Windows systems.
3. **IP Address Manipulation Libraries**: Libraries that handle IP address parsing, manipulation, and validation may include support for IPv6 addresses.
4. **DNS APIs**: DNS (Domain Name System) APIs may support IPv6 for resolving domain names to IPv6 addresses and vice versa.
5. **Network Management APIs**: APIs for network monitoring, diagnostics, and management may include support for IPv6-related functions. For example:
   * **SNMP (Simple Network Management Protocol)**: Used for network management and monitoring, includes support for IPv6 in newer versions.
   * **Net-SNMP library (C)**: A widely used SNMP library with support for IPv6.

**6Bone**

The 6bone was an experimental IPv6 network infrastructure that served as a global testbed for early IPv6 deployments and research. It was established in 1996 by the Internet Engineering Task Force (IETF) and operated until it was phased out in 2006. The name "6bone" is a play on words, combining "IPv6" with the term "backbone."

Key features and aspects of the 6bone include:

1. **Testing Ground for IPv6**: The primary purpose of the 6bone was to provide a platform for testing and experimenting with IPv6 technologies, protocols, and applications. It allowed researchers, network operators, and developers to gain practical experience with IPv6 before its widespread deployment in production networks.
2. **Global Reach**: The 6bone was a globally distributed network infrastructure, with participants and nodes located across multiple continents. It facilitated collaboration and knowledge sharing among the international IPv6 community.
3. **Transition Mechanisms**: The 6bone implemented various transition mechanisms to facilitate coexistence and interoperability between IPv4 and IPv6 networks. These mechanisms included tunneling, dual-stack operation, and translation technologies.
4. **Address Space**: The 6bone utilized a portion of the IPv6 address space reserved for experimental use (referred to as the "6bone address space"). Participants were assigned IPv6 addresses from this address space for use within the 6bone network.
5. **Coordination and Governance**: The 6bone was coordinated and governed by a community of volunteers, including network operators, researchers, and IPv6 enthusiasts. Coordination efforts were primarily conducted through mailing lists, forums, and coordination centers.
6. **Phased Decommissioning**: As IPv6 matured and became more widely deployed in production networks, the need for a separate experimental infrastructure like the 6bone diminished. Beginning in 2004, the 6bone was gradually phased out, and its resources were reclaimed for other purposes. The official decommissioning of the 6bone occurred on June 6, 2006.

Unit 3

**Mobility in Network**

Mobility in network refers to the ability of a device or user to move within a network while maintaining connectivity and communication. This concept is particularly relevant in wireless networks where devices can move freely within the coverage area. Mobility management protocols and techniques are essential for ensuring seamless connectivity as devices move.

Key aspects of mobility in networks include:

1. **Handover/Handoff**: When a mobile device moves from one network cell to another, it needs to seamlessly transfer its connection from one base station to another. This process is called handover or handoff. It ensures continuity of service without interruption.
2. **Roaming**: Roaming allows a mobile device to maintain connectivity and communication services while moving outside the coverage area of its home network. Roaming agreements between network operators enable users to access services while traveling.
3. **Location Management**: Location management involves keeping track of the current location of mobile devices within the network. This information is essential for efficient routing of data and for delivering services to mobile users.
4. **Quality of Service (QoS)**: Maintaining QoS for mobile users is crucial for delivering a satisfactory user experience. QoS parameters may include metrics such as latency, packet loss, and bandwidth availability, which need to be managed as devices move within the network.
5. **Mobility Protocols**: Various protocols are used to manage mobility in networks. For example, Mobile IP (Internet Protocol) enables mobile devices to maintain a consistent IP address even as they move between different networks.
6. **Security**: Ensuring the security of mobile communications is essential, especially in wireless networks where data transmissions are more vulnerable to interception. Security measures such as encryption and authentication help protect data as it moves between network nodes.

**Mobile Security**

Mobile security in computer networks refers to the measures and protocols put in place to protect data, devices, and networks from security threats in the context of mobile computing. With the proliferation of smartphones, tablets, and other portable devices, ensuring the security of data and communications has become increasingly important. Here are some key aspects of mobile security in computer networks:

1. **Device Security**: Securing the mobile device itself is essential. This involves implementing measures such as strong device passcodes or biometric authentication, keeping device operating systems and applications updated with the latest security patches, and enabling built-in security features such as remote wipe and Find My Device functionality.
2. **Data Encryption**: Encrypting sensitive data stored on the device helps protect it from unauthorized access in case the device is lost or stolen. Encryption should be applied to both data at rest (stored on the device) and data in transit (being transmitted over networks).
3. **Network Security**: Securing network communications is crucial, especially for devices that frequently connect to public Wi-Fi networks. This includes using Virtual Private Networks (VPNs) to encrypt internet traffic, avoiding unsecured Wi-Fi networks, and implementing protocols such as WPA2 for secure Wi-Fi connections.
4. **Mobile Device Management (MDM)**: MDM solutions enable centralized management and control of mobile devices within an organization. They allow administrators to enforce security policies, remotely configure devices, and ensure compliance with security standards.
5. **App Security**: Mobile apps can pose security risks, particularly if they are downloaded from untrusted sources or if they request excessive permissions. App security involves vetting and monitoring apps for security vulnerabilities, implementing app sandboxing to restrict their access to device resources, and educating users about safe app usage practices.
6. **Authentication and Authorization**: Implementing strong authentication mechanisms, such as two-factor authentication (2FA) or biometric authentication, helps ensure that only authorized users can access sensitive data and services on mobile devices.
7. **Mobile Threat Defense (MTD)**: MTD solutions are designed to detect and mitigate mobile-specific security threats, such as malware, phishing attacks, and network exploits targeting mobile devices. These solutions often employ techniques such as behavioral analysis, app reputation scanning, and network traffic monitoring.
8. **User Education and Awareness**: Educating users about common mobile security threats and best practices for staying safe, such as avoiding suspicious links and keeping devices updated, is essential for maintaining a secure mobile computing environment.

By implementing a combination of these measures, organizations can enhance the security posture of their mobile devices and networks, reducing the risk of data breaches, unauthorized access, and other security incidents.

* Top of Form

**IP Multicasting**

Multicasting that takes place over the Internet is known as IP Multicasting. These multicast follow the internet protocol(IP) to transmit data. IP multicasting uses a mechanism known as ‘Multicast trees’ to transmit to information among the users of the network. Multicast trees; allows a single transmission to branch out to the desired receivers. The branches are created at the Internet routers, the branches are created such that the length of the transmission will be minimum.

IP multicasts also use two other essential protocols to function; Internet Group Management Protocol (IGMP), Protocol Independent Multicast (PIM). IGMP allows the recipients to access the data or information i.e if any host wants to receive the message that is going to be multicasted, they must join the group using this protocol. The network routers use PIM to create multicast trees. To sum up, Multicasting is an efficient way of communication; it reduces the bandwidth usage and is used when a message is to be sent to a large number of selected individuals.

Here's how IP multicasting works:

1. **Sender**: The sender is the device that initiates the multicast transmission. It encapsulates the data to be multicast into IP packets and specifies a multicast IP address as the destination address for the packets.
2. **Multicast IP Address**: Unlike unicast communication, where data is sent to a specific IP address representing a single device, multicast communication uses special IP addresses known as multicast group addresses. These addresses are in the range of 224.0.0.0 to 239.255.255.255. Devices interested in receiving multicast traffic join specific multicast groups by subscribing to the corresponding multicast IP addresses.
3. **Routing**: Multicast packets are routed through the network based on multicast routing protocols such as Protocol Independent Multicast (PIM) and Internet Group Management Protocol (IGMP). Routers in the network use these protocols to efficiently forward multicast packets to only those network segments where receivers have expressed interest in the multicast group.
4. **Receivers**: Receivers are devices that are part of a multicast group and are interested in receiving the multicast traffic. They join the multicast group by sending IGMP messages to their local router, indicating their interest in receiving traffic destined for the multicast group address.
5. **Traffic Replication**: Routers in the network replicate multicast packets only when necessary, based on the presence of receivers on different network segments. This reduces network bandwidth usage compared to sending separate unicast packets to each receiver.

**Multicast Routing Protocol**

Multicast routing protocols are specifically designed to efficiently route multicast traffic in IP networks. There are several multicast routing protocols, each serving different network environments and requirements. Here are some common multicast routing protocols:

1. **Protocol Independent Multicast (PIM)**:
   * PIM is the most widely used multicast routing protocol in IP networks.
   * It operates independently of the underlying unicast routing protocol, making it suitable for various network environments.
   * PIM comes in two main modes: PIM Sparse Mode (PIM-SM) and PIM Dense Mode (PIM-DM).
   * PIM-SM is typically used in sparse multicast environments, where multicast traffic is sporadic and receivers are widely dispersed.
   * PIM-DM is suitable for dense multicast environments, where multicast traffic is abundant and receivers are densely located.
2. **Multicast Open Shortest Path First (MOSPF)**:
   * MOSPF is an extension of the OSPF unicast routing protocol to support multicast routing.
   * It operates by building multicast distribution trees based on OSPF routing information.
   * MOSPF is less commonly used compared to PIM, and its usage is mainly restricted to environments where OSPF is the primary unicast routing protocol.
3. **Distance Vector Multicast Routing Protocol (DVMRP)**:
   * DVMRP is one of the earliest multicast routing protocols developed for IP networks.
   * It operates based on a distance-vector algorithm, similar to traditional unicast routing protocols like RIP.
   * DVMRP is primarily used in legacy networks or environments where other multicast routing protocols are not feasible.
4. **Bidirectional PIM (Bidir-PIM)**:
   * Bidir-PIM is an extension of PIM designed specifically for bidirectional multicast traffic.
   * It establishes shared distribution trees for both upstream and downstream traffic flow, reducing the complexity of managing bidirectional multicast sessions.
5. **Source-Specific Multicast (SSM)**:
   * SSM is not a routing protocol but rather a service model for multicast traffic delivery.
   * It allows receivers to specify the exact source of multicast traffic they are interested in, simplifying the multicast routing process.
   * SSM can be implemented using PIM-SM or other multicast routing protocols with source filtering capabilities.

**Address Assignment**

Address assignment in networking refers to the process of assigning unique identifiers to devices on a network so that they can communicate with each other. There are several methods for address assignment, depending on the type of network and the specific requirements. Here are some common methods:

1. **Static IP Address Assignment**:
   * In this method, a network administrator manually assigns a fixed IP address to each device on the network.
   * Static IP addresses are typically used for servers, network printers, and other devices that require permanent, predictable addresses.
   * It provides stability and control but can be time-consuming to manage, especially in large networks.
2. **Dynamic Host Configuration Protocol (DHCP)**:
   * DHCP is a network protocol used to automatically assign IP addresses and other network configuration parameters to devices on a network.
   * A DHCP server dynamically assigns IP addresses from a pool of available addresses to requesting devices.
   * DHCP simplifies address management and reduces the risk of address conflicts.
   * It is widely used in home and enterprise networks.
3. **Automatic Private IP Addressing (APIPA)**:
   * APIPA is a feature of Windows operating systems that automatically assigns IP addresses to devices when a DHCP server is not available.
   * Devices configured for APIPA use a predefined range of IP addresses (169.254.0.0/16) and self-assign an address if they cannot obtain one from a DHCP server.
   * APIPA allows devices to communicate with each other within a local network segment, but not with devices on other subnets.
4. **IPv6 Address Autoconfiguration**:
   * IPv6 hosts can automatically configure their IP addresses using a combination of stateless address autoconfiguration (SLAAC) and DHCPv6.
   * With SLAAC, devices generate their IPv6 addresses using the network prefix advertised by routers and their own unique interface identifier.
   * DHCPv6 can be used to provide additional network configuration parameters, such as DNS server addresses and domain information.
5. **Manual Configuration**:
   * Some networks may require manual configuration of IP addresses, particularly in specialized or legacy environments.
   * Network administrators manually assign IP addresses, subnet masks, default gateways, and other configuration parameters to devices.
   * Manual configuration provides precise control but can be prone to errors and is less scalable than dynamic methods.

**Session Discovery**

Session discovery in computer networks typically refers to the process by which devices on a network identify and establish communication sessions with each other. These sessions are temporary connections established for the purpose of exchanging data between devices.

Here's a general overview of how session discovery works:

1. **Network Discovery**: Devices on a network need to discover each other before they can establish communication sessions. This can be achieved through various network discovery protocols or methods such as ARP (Address Resolution Protocol), ICMP (Internet Control Message Protocol), DNS (Domain Name System), or service discovery protocols like SSDP (Simple Service Discovery Protocol) or Bonjour.
2. **Address Resolution**: Once a device is discovered, the next step is to resolve its network address. For example, in IP-based networks, this involves mapping a device's logical address (such as an IP address) to its physical address (such as a MAC address) using protocols like ARP.
3. **Session Establishment**: Once the network address of the target device is resolved, the initiating device can then establish a session with it. This typically involves a negotiation process where the two devices agree on parameters such as communication protocols, encryption methods, session timeouts, etc.
4. **Session Maintenance**: During the session, both devices involved in communication need to maintain state information to ensure the integrity and continuity of the session. This includes monitoring the session for errors, managing resources allocated to the session, and handling any changes in network conditions.
5. **Session Termination**: When the communication session is no longer needed, it is terminated either explicitly by one of the devices or implicitly due to timeout or network failure. Proper session termination is important to release resources and maintain network efficiency.

**tcp extension for high speed networks**

In high-speed networks, the traditional TCP (Transmission Control Protocol) may encounter performance limitations due to its conservative approach to congestion control and error recovery. To address these limitations, several TCP extensions and variants have been proposed and developed. These extensions aim to improve the efficiency and performance of TCP in high-speed networks. Some of these TCP extensions include:

1. **TCP Vegas**: TCP Vegas is a congestion control algorithm designed to improve TCP's performance in high-speed and high-bandwidth networks. It uses a different approach to congestion avoidance compared to traditional TCP's additive increase/multiplicative decrease mechanism. Vegas aims to detect congestion before packet loss occurs by monitoring the rate of change of the Round-Trip Time (RTT). By reacting to changes in RTT, Vegas can adjust the sending rate more accurately and avoid unnecessary packet loss.
2. **TCP NewReno**: TCP NewReno is an extension of the TCP Reno congestion control algorithm. It addresses some of the limitations of TCP Reno related to fast recovery after multiple packet losses. NewReno improves TCP's performance in high-speed networks by enhancing its fast recovery mechanism to allow for more aggressive retransmission of lost packets.
3. **TCP SACK (Selective Acknowledgment)**: TCP SACK is an extension that enhances TCP's error recovery mechanism by allowing the receiver to selectively acknowledge out-of-order segments. This enables the sender to retransmit only the missing segments instead of retransmitting the entire window, which can be more efficient in high-speed networks with a high likelihood of packet reordering.
4. **TCP ECN (Explicit Congestion Notification)**: TCP ECN is an extension that allows routers to notify endpoints of impending congestion before packet loss occurs. This helps TCP to react more quickly to congestion without waiting for packet loss signals. ECN can be beneficial in high-speed networks where congestion can occur more frequently but may not always lead to packet loss.
5. **TCP Fast Open**: TCP Fast Open is an extension that allows data to be exchanged during the TCP handshake process, reducing the latency associated with establishing new connections. This can improve the performance of short-lived connections, such as those encountered in web browsing, particularly in high-speed networks where latency is a critical factor.

**Transaction Oriented Application**

A transaction-oriented application is a type of software system designed to manage transactions reliably and efficiently. A transaction typically refers to a sequence of operations performed by an application or user that must be executed as a single unit of work. These operations often involve accessing and modifying data in a database or performing other related tasks. Transaction-oriented applications ensure that these operations are executed reliably and consistently, even in the presence of failures or concurrent access by multiple users.

Here are some key characteristics and considerations of transaction-oriented applications:

1. **Atomicity**: Transactions in transaction-oriented applications are atomic, meaning they are either fully completed or fully aborted. This ensures that all operations within a transaction are either applied entirely or not at all, maintaining the consistency of the data.
2. **Consistency**: Transaction-oriented applications maintain data consistency by enforcing constraints and integrity rules defined by the application or database schema. Transactions ensure that data remains in a valid state throughout the execution of operations.
3. **Isolation**: Transactions are often executed concurrently in multi-user environments. Isolation ensures that each transaction appears to execute in isolation from other transactions, even though they may be executing concurrently. This prevents interference and maintains data integrity.
4. **Durability**: Once a transaction is committed, its changes are durable and persistent, even in the event of system failures. Durability is typically achieved through mechanisms such as logging and recovery techniques to ensure that committed transactions survive system crashes.
5. **Concurrency Control**: Transaction-oriented applications implement concurrency control mechanisms to manage concurrent access to shared resources, such as databases, to prevent conflicts and maintain data consistency. Techniques like locking, optimistic concurrency control, and multiversion concurrency control are commonly used.
6. **ACID Properties**: Transaction-oriented applications adhere to the ACID properties: Atomicity, Consistency, Isolation, and Durability. These properties ensure that transactions maintain data integrity and reliability.

Transaction-oriented applications are commonly used in various domains, including banking and finance, e-commerce, inventory management, and enterprise resource planning (ERP) systems. Examples of transaction-oriented systems include online banking applications, point-of-sale systems, and reservation systems.

Implementing transaction-oriented applications requires careful design and consideration of factors such as data consistency requirements, concurrency control mechanisms, and fault tolerance. Additionally, choosing the appropriate technologies and architectures, such as relational databases, distributed systems frameworks, and messaging systems, is crucial for building reliable and scalable transaction-oriented applications.

Unit 4

Network security operates at multiple layers of the OSI (Open Systems Interconnection) model, each layer providing different mechanisms and protocols to protect data and systems from various threats. Here's an overview of network security measures at different layers:

1. **Physical Layer (Layer 1)**:
   * Physical security measures such as locked server rooms, surveillance cameras, and biometric access controls protect physical network components from unauthorized access.
   * Cable shielding and encryption
2. **Data Link Layer (Layer 2)**:
   * MAC address filtering allows only authorized devices to communicate on the network.
   * Virtual LANs (VLANs) segment the network to restrict communication between devices.
   * Switch port security limits the number of MAC addresses that can access a switch port.
3. **Network Layer (Layer 3)**:
   * Firewalls filter traffic based on IP addresses, ports, and protocols to enforce network policies and prevent unauthorized access.
   * Virtual Private Networks (VPNs) encrypt data between networks or remote users, ensuring confidentiality and integrity.
   * Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) detect and block malicious network activities.
4. **Transport Layer (Layer 4)**:
   * Transport layer security protocols like SSL/TLS provide encryption and authentication for data exchanged between endpoints.
   * Transport layer firewalls inspect and filter traffic based on port numbers and protocol headers.
5. **Session Layer (Layer 5)**:
   * Secure Sockets Layer (SSL) and Transport Layer Security (TLS) establish secure sessions between applications, ensuring data confidentiality and integrity.
   * Session management mechanisms authenticate and authorize users before establishing connections.
6. **Presentation Layer (Layer 6)**:
   * Encryption mechanisms such as data encryption standards (DES), Advanced Encryption Standard (AES), and RSA protect data during transmission and storage.
   * Data compression reduces bandwidth usage and enhances performance.
7. **Application Layer (Layer 7)**:
   * Application-level firewalls inspect and filter traffic based on application-layer protocols and content, protecting against application-layer attacks.
   * Authentication mechanisms such as multi-factor authentication (MFA) and single sign-on (SSO) verify user identities and control access to applications.
   * Antivirus and antimalware software detect and remove malicious software from files and applications.
   * Web application firewalls (WAFs) protect web applications from common vulnerabilities and attacks such as SQL injection and cross-site scripting (XSS).

**HTTPS**

**HTTPS** stands for **Hyper Text Transfer Protocol Secure**. HTTP Secure (HTTPS), could be a combination of the Hypertext Transfer Protocol with the SSL/TLS convention to supply encrypted communication and secure distinguishing proof of a arrange web server.   
If the URL of that site is just HTTP, at that point anything you’re perusing or whatever points of interest you’re putting on that site, on the off chance that a programmer needs to take your data.   
Therefore, HTTPS is more secure than HTTP because HTTPS is certified by the SSL(Secure Socket Layer). Whatever website you are visiting on the internet, if its URL is HTTP, then that website is not secure.   
If a website has an SSL certificate installed then the URL of that website will be HTTPS that website will completely secure. You can give any information about your credit card, debit cards, OTP and anything else.

**Characteristics of HTTPS:**

* **Security**: Nowadays there’s a lot of cyber-attacks on the web. And online installments have also expanded. That’s why we need to be secure.If there is no security in any website,then no will use that website.
* **Need of SSL**: Some SEO specialists accept that by introducing SSL on the site, there are a few SEO benefits from Google. And by applying SSL, the positioning of the site in Google is additionally boosted.
* **Authentication**: HTTPS encrypts all message substance, including the HTTP headers and the request/response data.The verification perspective of HTTPS requires a trusted third party to sign server-side digital certificates.
* **Browsing Privately**: HTTPS is presently utilised more frequently by web clients than the first non-secure HTTP, fundamentally to ensure page genuineness on all sorts of websites,secure accounts and to keep client communications.

**Advantages of HTTPS:**

* Secures your information in-transit.
* Help you boost income per client.
* Protects your site from Phishing, MITM and other information breaches.
* Builds believe on your site visitors. Removes “NOT Secure” warnings.
* Help you move forward website ranking.

**Disadvantages of HTTPS:**

* A web ask with HTTPS is slower which regularly comes about in moderate page stacking.
* Pages with HTTPS can never be cached could be a shared cache.
* A few intermediary serves or firewall frameworks don’t permit get to to locales with HTTPS.
* If you’re making web site which has static contents or if there’s no private information exchange, you’ll select the HTTP.

**SSL**

SSL (Secure Sockets Layer) is a cryptographic protocol designed to provide secure communication over a computer network, typically the internet. SSL ensures that data transmitted between a client and a server is encrypted and remains confidential, and it also verifies the identity of the server to the client.

SSL operates at the transport layer of the OSI (Open Systems Interconnection) model and is used to secure various application-layer protocols, including HTTP (to create HTTPS), FTP, SMTP, and IMAP.

SSL has largely been replaced by its successor, Transport Layer Security (TLS), but the term "SSL" is still commonly used to refer to both SSL and TLS protocols.

SSL provides security to confidential data following the encryption process over the internet. To provide data privacy, secure connection between the server and the browser is created. The SSL-certified website contains a padlock before the URL of the website, which indicates that the website is secure to perform the transactions.

To ensure data integrity, privacy and authentication, in 1995 Netscape developed an encryption-based Internet security protocol called SSL, Secure Sockets Layer.

**Advantages**

* **Security:** The main goal of SSL is to provide security to the data or information by encryption. The data traveling through the internet is always at high risk of getting lost. Thus, SSL encrypts the data so that even if the data is received by a third party they will not understand it.
* **Authentication:** When the data travels on the internet, it has chances to get accessed by a third party. To ensure authentication the SSL uses a Server certificate that makes sure that the SSL certificate provider is an authorized one to establish the connection between the SSL server and browser.
* **Reliability:** The SSL certificate verifies to the visitor if the website visited is legitimate and not fake. Thus, the visitor can easily trust and rely on the website.
* **Software Requirements:**SSL doesn’t require any specific client software to be installed. It only needs an internet connection and web browser. Thus, in this manner it can be cost effective.

**Disadvantages**

* **Performance:**When a site uses the SSL certificate, the data transaction speed gets affected. As the data gets encrypted and decrypted, the performance goes down.
* **Expiry:** The SSL certificate needs to be renewed from time to time. If it is not renewed then the performance gets affected.
* **Application Support:** Initially, the SSL was meant to support web-based applications. In addition, the setup process is complex.

**ESP**

**Encapsulating Security Payload (ESP**) provides all encryption services in IPSec based on integrity for the payload and not for the IP header, confidentiality and authentication that using encryption, without authentication is strongly discouraged because it is insecure.

Any translations in readable message format into an unreadable format are encrypted and used to hide the message content against data tampering.

IPSec provides an open framework, such as SHA and MD5 for implementing industry standard algorithms.

Encryption/decryption allows only the sender and the authorised receiver to make the data to be received in readable form and only after the integrity verification process is complete, the data payload in the packet is decrypted.

IPSec uses a unique identifier for each packet, which is a data equivalent of a fingerprint and checks for packets that are authorised or not. It doesn't sign the entire packet unless it is being tunnelled—ordinarily, for this IP data payload is protected, not the IP header. In Tunnel Mode, where the entire original IP packet is encapsulated with a new packet header added.

ESP in transport mode does not provide integrity and authentication for the entire IP packet.

**Modes in ESP:**

Encapsulating Security Payload supports two modes, i.e. Transport mode, and tunnel mode.

**Tunnel mode:**

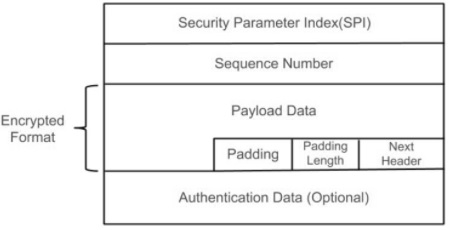
1. Mandatory in Gateway, tunnel mode holds utmost importance.
2. Here, a new IP Header is created which is used as the outer IP Header followed by ESP.

**Transport mode:**

1. Here, IP Header is not protected via encryption or authentication, making it vulnerable to threats
2. Less processing is seen in this mode, so the inclusion of ESP is preferred

**ESP Format**

The ESP format is diagrammatically represented as follows –



**Explanation**

* **Security Parameters Index (32 bits)** − Identifies a security association. This field is mandatory. The value of zero is reserved for local, implementation- specific use and MUST NOT be sent on the wire.
* **Sequence Number (32 bits)** − A monotonically increasing counter value; this provides an anti-replay function, as discussed for AH. The first packet sent using a given SA will have a Sequence number of 1.
* **Payload Data (variable)** − This is a transport-level segment (transport mode) or IP packet (tunnel mode) that is protected by encryption. The type of content that was protected is indicated by the Next Header field.
* **Padding (0-255 bytes) −** Padding for encryption, to extend the payload data to a size that fits the encryption's cipher block size, and to align the next field.
* **Pad Length (8 bits) −** Indicates the number of pad bytes immediately preceding this field.
* **Next Header (8 bits) −** Identifies the type of data contained in the payload data field by identifying the first header in that payload.
* **Authentication Data (variable)** − A variable-length field (must be an integral number of 32-bit words) that contains the Integrity. Check Value computed over the ESP packet minus the Authentication Data field. This field is optional and is included only if the authentication service has been selected for the SA in question.

**Authentication Format**

Authentication Header (AH) is used to provide integrity and authentication to IP datagrams. Replay protection is also possible. The services are connectionless, that means they work on a per-packet basis.

AH is used in two modes as follows −

* Transport mode
* Tunnel mode

AH authenticates are the same as IP datagram. In transport mode, some fields in the IP header change en-route and their value cannot be predicted by the receiver. These fields are called mutable and they are not protected by AH.

**Mutable IPv4 fields**

The mutable IPv4 fields are as follows −

* Type of service (TOS)
* Flags
* Fragment offset
* Time to live (TTL)
* Header checksum

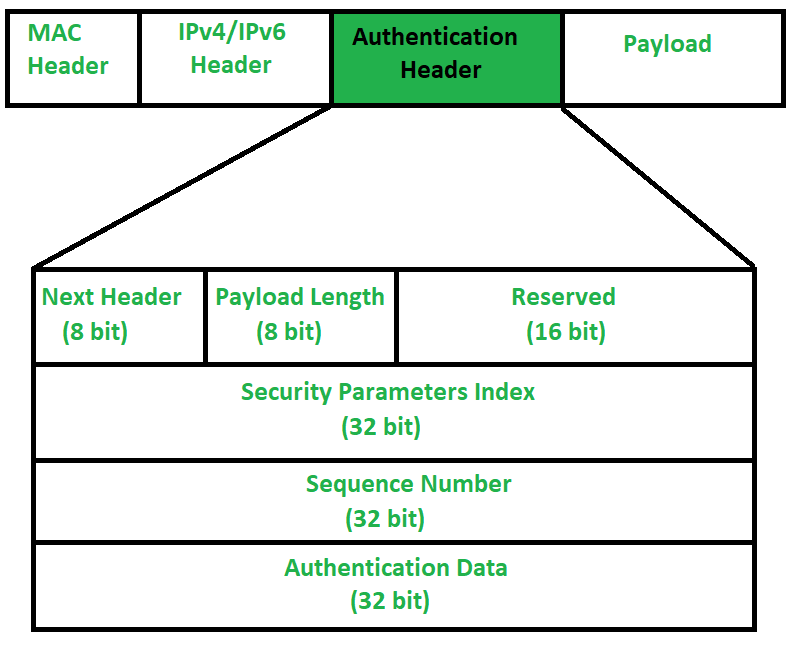
To protect these fields, tunnelling must be used. The payload of the IP packet is considered immutable and is always protected by AH. AH processing is applied only to non-fragmented IP packets. Whereas an IP packet with AH applied can be fragmented by intermediate routers.

In this case, the destination first reassembles the packet and then applies AH processing to it. If an IP packet that appears to be a fragment is input to AH processing, and it is discarded.

This prevents the overlapping fragment attack, which misuses the fragment reassembly algorithm to create forged packets and force them through a firewall. Packets that fail authentication are discarded and never delivered to upper layers. This mode of operation greatly reduces the chances of successful denial-of-service attacks.

**AH format**

The AH format is described in RFC 2402. The below shows the position of the Authentication Header fields in the IP packet.



The fields are as follows −

1. **Next Header –**Next Header is 8-bit field that identifies type of header present after Authentication Header. In case of TCP, UDP or destination header or some other extension header it will store correspondence IP protocol number . Like, number 4 in this field will indicate IPv4, number 41 will indicate IPv6 and number 6 will indicate TCP.
2. **Payload Length** – Payload length is length of Authentication header and here we use scaling factor of 4. Whatever be size of header, divide it by 4 and then subtract by 2. We are subtracting by 2 because we’re not counting first 8 bytes of Authentication header, which is first two row of picture given above. It means we are not including Next Header, Payload length, Reserved and Security Parameter index in calculating payload length. Like, say if payload length is given to be X. Then (X+2)\*4 will be original Authentication header length.
3. **Reserved** – This is 16-bit field which is set to “zero” by sender as this field is reserved for future use.
4. **Security Parameter Index (SPI)** – It is arbitrary 32-bit field. It is very important field which identifies all packets which belongs to present connection. If we’re sending data from Source A to Destination B. Both A and B will already know algorithm and key they are going to use. So for Authentication, hashing function and key will be required which only source and destination will know about. Secret key between A and B is exchanged by method of Diffie Hellman algorithm. So Hashing algorithm and secret key for Security parameter index of connection will be fixed. Before data transfer starts security association needs to be established. In Security Association, both parties needs to communicate prior to data exchange. Security association tells what is security parameter index, hashing algorithm and secret key that are being used.
5. **Sequence Number** – This unsigned 32-bit field contains counter value that increases by one for each packet sent. Every packet will need sequence number. It will start from 0 and will go till 223 – 1 and there will be no wrap around. Say, if all sequence numbers are over and none of it is left but we cannot wrap around as it is not allowed. So, we will end connection and re-establish connection again to resume transfer of remaining data from sequence number 0. Basically, sequence numbers are used to stop replay attack. In Replay attack, if same message is sent twice or more, receiver won’t be able to know if both messages are sent from a single source or not. Say, I am requesting 100$ from receiver and Intruder in between asked for another 100$. Receiver won’t be able to know that there is intruder in between.
6. **Authentication Data (Integrity Check Value)** – Authentication data is variable length field that contains Integrity Check Value (ICV) for packet. Using hashing algorithm and secret key, sender will create message digest which will be sent to receiver. Receiver on other hand will use same hashing algorithm and secret key. If both message digest matches then receiver will accept data. Otherwise, receiver will discard it by saying that message has been modified in between. So basically, authentication data is used to verify integrity of transmission. Also, length of Authentication data depends upon hashing algorithm you choose.

**Key Distribution Protocol**

Key distribution protocols are essential in cryptographic systems for securely and efficiently distributing cryptographic keys between parties involved in secure communication. These protocols facilitate the establishment of secure communication channels and the encryption and decryption of messages. Here are some key distribution protocols commonly used in cryptographic systems:

1. **Diffie-Hellman Key Exchange (DH)**:
   * Diffie-Hellman is a key exchange protocol that allows two parties to establish a shared secret key over an insecure communication channel without prior shared knowledge.
   * The protocol is based on the discrete logarithm problem, where both parties agree on common parameters (prime modulus and generator) and perform mathematical operations to derive a shared secret key.
2. **RSA (Rivest-Shamir-Adleman)**:
   * RSA is a public-key encryption algorithm that can also be used for key distribution.
   * In RSA key exchange, one party generates a public/private key pair and shares the public key with the other party. The other party can then encrypt a secret key using the public key and send it securely to the first party, who decrypts it using their private key.
3. **Elliptic Curve Cryptography (ECC)**:
   * ECC is another asymmetric encryption algorithm that is increasingly popular due to its efficiency and smaller key sizes compared to RSA.
   * ECC key exchange protocols, such as Elliptic Curve Diffie-Hellman (ECDH), operate similarly to DH but use elliptic curve mathematics for key generation and exchange.
4. **Key Distribution Center (KDC)**:
   * KDC is a centralized service that facilitates key distribution in symmetric-key cryptography systems, such as Kerberos.
   * In Kerberos, the KDC issues tickets containing session keys to authenticated users, which they can use to establish secure communication with network services.
5. **Transport Layer Security (TLS)**:
   * TLS is a widely used protocol for securing communication over the internet, commonly used in web browsers for HTTPS.
   * TLS employs a combination of symmetric and asymmetric cryptography for key exchange, authentication, and encryption, with key exchange protocols such as Diffie-Hellman or RSA being used during the TLS handshake process.
6. **Secure Shell (SSH)**:
   * SSH is a protocol used for secure remote login and file transfer over an insecure network.
   * SSH employs various key exchange algorithms, including Diffie-Hellman, RSA, and Elliptic Curve Diffie-Hellman (ECDH), to establish secure communication channels between clients and servers.

**Digital Signature**

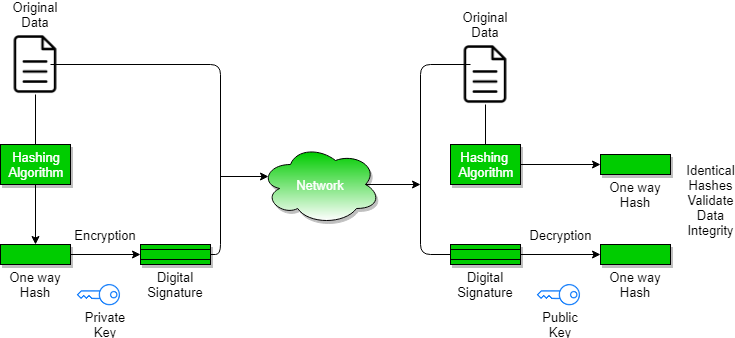
A digital signature is a mathematical technique used to validate the authenticity and integrity of a message, software, or digital document. 

1. **Key Generation Algorithms**: Digital signature is electronic signatures, which assure that the message was sent by a particular sender. While performing digital transactions authenticity and integrity should be assured, otherwise, the data can be altered or someone can also act as if he was the sender and expect a reply.
2. **Signing Algorithms**: To create a digital signature, signing algorithms like email programs create a one-way hash of the electronic data which is to be signed. The signing algorithm then encrypts the hash value using the private key (signature key). This encrypted hash along with other information like the hashing algorithm is the digital signature. This digital signature is appended with the data and sent to the verifier. The reason for encrypting the hash instead of the entire message or document is that a hash function converts any arbitrary input into a much shorter fixed-length value. This saves time as now instead of signing a long message a shorter hash value has to be signed and moreover hashing is much faster than signing.
3. **Signature Verification Algorithms** : Verifier receives Digital Signature along with the data. It then uses Verification algorithm to process on the digital signature and the public key (verification key) and generates some value. It also applies the same hash function on the received data and generates a hash value. If they both are equal, then the digital signature is valid else it is invalid.

**The steps followed in creating digital signature are :**

1. Message digest is computed by applying hash function on the message and then message digest is encrypted using private key of sender to form the digital signature. (digital signature = encryption (private key of sender, message digest) and message digest = message digest algorithm(message)).
2. Digital signature is then transmitted with the message.(message + digital signature is transmitted)
3. Receiver decrypts the digital signature using the public key of sender.(This assures authenticity, as only sender has his private key so only sender can encrypt using his private key which can thus be decrypted by sender’s public key).
4. The receiver now has the message digest.
5. The receiver can compute the message digest from the message (actual message is sent with the digital signature).
6. The message digest computed by receiver and the message digest (got by decryption on digital signature) need to be same for ensuring integrity.

Message digest is computed using one-way hash function, i.e. a hash function in which computation of hash value of a message is easy but computation of the message from hash value of the message is very difficult.



**Assurances about digital signatures**

The definitions and words that follow illustrate the kind of assurances that digital signatures offer.

1. **Authenticity**: The identity of the signer is verified.
2. **Integration:** Since the content was digitally signed, it hasn’t been altered or interfered with.
3. **Non-repudiation:**demonstrates the source of the signed content to all parties. The act of a signer denying any affiliation with the signed material is known as repudiation.
4. **Notarization:**Under some conditions, a signature in a Microsoft Word, Microsoft Excel, or Microsoft PowerPoint document that has been time-stamped by a secure time-stamp server is equivalent to a notarization.

**Digital Certificate**

Digital certificate is issued by a trusted third party which proves sender’s identity to the receiver and receiver’s identity to the sender.   
A digital certificate is a certificate issued by a Certificate Authority (CA) to verify the identity of the certificate holder. Digital certificate is used to attach public key with a particular individual or an entity.

**Digital certificate contains**

* Name of certificate holder.
* Serial number which is used to uniquely identify a certificate, the individual or the entity identified by the certificate
* Expiration dates.
* Copy of certificate holder’s public key.(used for decrypting messages and digital signatures)
* Digital Signature of the certificate issuing authority.
* Digital certificate is also sent with the digital signature and the message.

**Advantages of Digital Certificate**

* **NETWORK SECURITY :**A complete, layered strategy is required by modern cybersecurity methods, wherein many solutions cooperate to offer the highest level of protection against malevolent actors. An essential component of this puzzle is digital certificates, which offer strong defence against manipulation and man-in-the-middle assaults.
* **VERIFICATION :**Digital certificates facilitate cybersecurity by restricting access to sensitive data, which makes authentication a crucial component of cybersecurity. Thus, there is a decreased chance that hostile actors will cause chaos. At many different endpoints, certificate-based authentication provides a dependable method of identity verification. Compared to other popular authentication methods like biometrics or one-time passwords, certificates are more flexible.
* **BUYER SUCCESS :**Astute consumers demand complete assurance that the websites they visit are reliable. Because digital certificates are supported by certificate authority that users’ browsers trust, they offer a readily identifiable indicator of reliability.

**Disadvantages of Digital Certificate**

* **Phishing attacks:** To make their websites look authentic, attackers can fabricate bogus websites and obtain certificates. Users may be fooled into providing sensitive information, such as their login credentials, which the attacker may then take advantage of.
* **Weak encryption:**Older digital certificate systems may employ less secure encryption methods that are open to intrusions.
* **Misconfiguration:** In order for digital certificates to work, they need to be set up correctly. Websites and online interactions can be attacked due to incorrectly configured certificates.

**Digital certificate vs digital signature**

Digital signature is used to verify authenticity, integrity, non-repudiation ,i.e. it is assuring that the message is sent by the known user and not modified, while digital certificate is used to verify the identity of the user, maybe sender or receiver. Thus, digital signature and certificate are different kind of things but both are used for security. Most websites use digital certificate to enhance trust of their users

| **Feature** | **Digital Signature** | **Digital Certificate** |
| --- | --- | --- |
| **Basics / Definition** | A digital signature secures the integrity of a digital document in a similar way as a fingerprint or attachment. | Digital certificate is a file that ensures holder’s identity and provides security. |
| **Process / Steps** | Hashed value of original data is encrypted using sender’s private key to generate the digital signature. | It is generated by CA (Certifying Authority) that involves four steps: Key Generation, Registration, Verification, Creation. |
| **Security Services** | **Authenticity** of Sender, **integrity** of the document and **non-repudiation**. | It provides security and **authenticity** of certificate holder. |
| **Standard** | It follows Digital Signature Standard (DSS). | It follows X.509 Standard Format |