Department of Information Technology

Question Bank for Unit 1

1. **What is data communication?**

   Data communication refers to the exchange of data between devices or systems through a transmission medium, such as cables, wireless channels, or optical fibers.

2. **What are the types of transmission modes in data communication?**

   The three types of transmission modes are:

   - **Simplex:** Data flows in one direction only.
   - **Half-duplex:** Data flows in both directions, but not at the same time.
   - **Full-duplex:** Data flows in both directions simultaneously.

3. **Define bandwidth.**

   Bandwidth is the maximum rate of data transfer over a communication channel, usually measured in bits per second (bps).

4. **What is a communication channel?**

   A communication channel is the medium through which data is transmitted from sender to receiver, such as copper wire, fiber optics, or wireless channels.

5. **What are the different types of networks?**

   The main types of networks are:

   - **LAN (Local Area Network)**
   - **MAN (Metropolitan Area Network)**
   - **WAN (Wide Area Network)**

OSI Model & TCP/IP Model

6. **What is the purpose of the OSI model?**

   The OSI model standardizes the functions of a communication system into seven distinct layers, making it easier to design and troubleshoot network systems.

7. **List the seven layers of the OSI model.**

The seven layers are:

1. Physical
2. Data Link
3. Network
4. Transport
5. Session
6. Presentation
7. Application

8. **What is the main difference between TCP and UDP?**

TCP (Transmission Control Protocol) is connection-oriented and guarantees reliable delivery of data, while UDP (User Datagram Protocol) is connectionless and does not guarantee reliable delivery.

9. **What does the Transport Layer in the OSI model do?**

The Transport Layer is responsible for ensuring reliable data transfer between two systems, providing error control, flow control, and segmentation.

10. **What is the function of the Network Layer in the OSI model?**

The Network Layer is responsible for routing data across networks using logical addresses (IP addresses) and determining the best path for data delivery.

Sockets & Protocol Layering

11. **What is a socket in networking?**

A socket is an endpoint for sending or receiving data across a network. It is used in client-server communication to establish a connection between the client and the server.

12. **What is the difference between a client socket and a server socket?**

A client socket initiates the connection to a server, while a server socket listens for incoming connections from clients.

13. **What is encapsulation in the OSI model?**

Encapsulation is the process of adding headers (and sometimes trailers) to data as it moves down the layers of the OSI model before being transmitted.

14. **What is the role of the Application Layer in the OSI model?**

The Application Layer provides network services directly to the user and supports protocols like HTTP, FTP, and SMTP for various application-level functions.

15. **What is the difference between IPv4 and IPv6?**

IPv4 uses 32-bit addresses, allowing for 4.3 billion unique addresses, while IPv6 uses 128-bit addresses, allowing for a vastly larger number of unique addresses.

Application Layer Protocols

16. **What is HTTP, and how does it work?**

HTTP (HyperText Transfer Protocol) is used for transferring hypertext documents on the web. It works by sending a request from the client (browser) to the server and receiving a response with the requested data.

17. **What is the difference between HTTP and HTTPS?**

HTTPS is the secure version of HTTP, using SSL/TLS encryption to secure data transfer between the client and server.

18. **What is FTP used for?**

FTP (File Transfer Protocol) is used for transferring files between computers over a network, allowing users to upload or download files to/from a server.

19. **What is the purpose of the SMTP protocol?**

SMTP (Simple Mail Transfer Protocol) is used to send and relay outgoing emails from a client to a mail server or between mail servers.

20. **What is the function of the DNS protocol?**

DNS (Domain Name System) is used to translate human-readable domain names (like www.example.com) into IP addresses required for locating resources on the internet.

1. Discuss the OSI Model in detail. Explain the functions of each layer.

The **OSI Model** (Open Systems Interconnection Model) is a conceptual framework used to understand and design network communication by breaking it down into seven layers. Here's a detailed description of each layer:

OSI Model Layers:
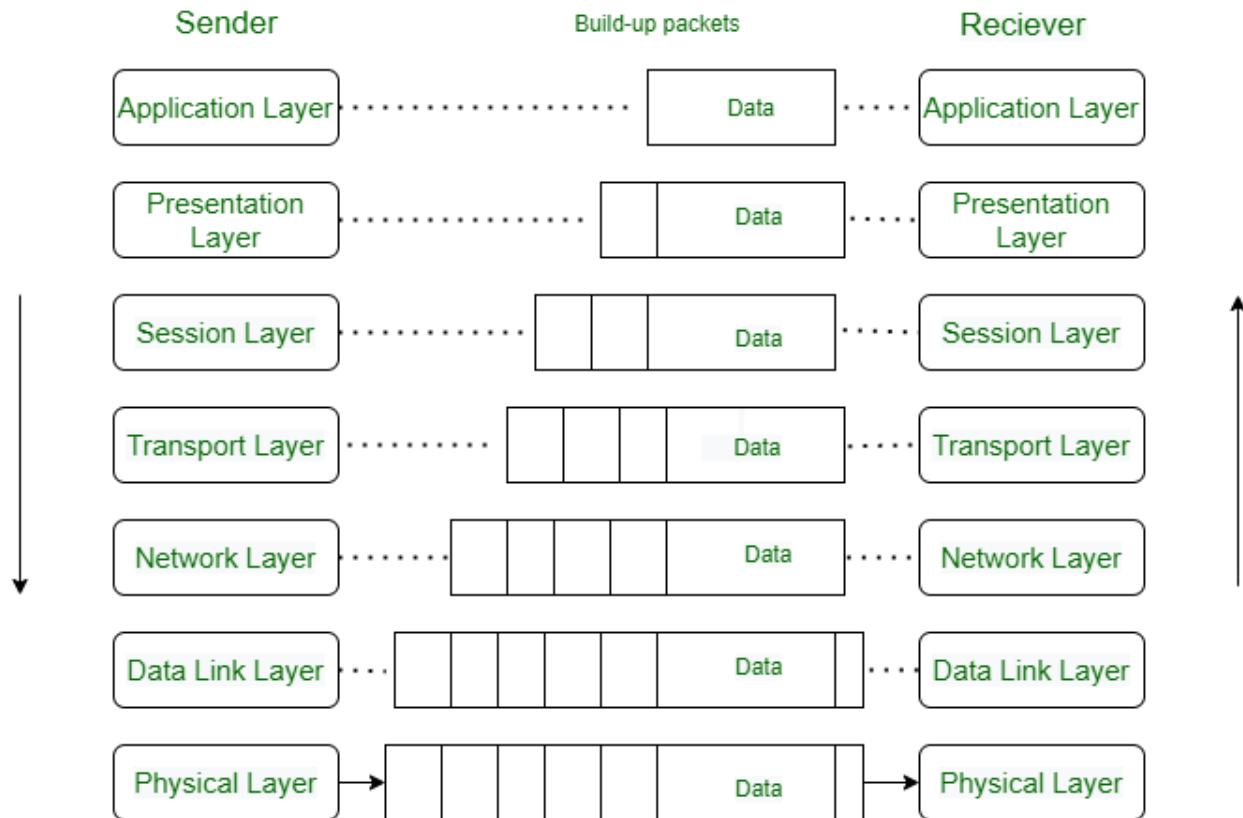
1. **Physical Layer**:
   - **Function**: Handles the transmission of raw bits over a physical medium like cables, radio signals, or optical fibers. It defines hardware specifications such as cables, switches, and signals.
   - **Example**: Cables, Hubs.
2. **Data Link Layer**:
   - **Function**: Provides error detection and correction, and organizes bits into frames for transmission over the physical medium.
   - **Example**: Ethernet, MAC addresses, Switches.
3. **Network Layer**:
   - **Function**: Manages the routing of data packets from the source to the destination device. Handles logical addressing (IP addresses) and routing.
   - **Example**: IP (Internet Protocol), Routers.

4. **Transport Layer**:
   - o **Function**: Ensures reliable data transfer between end-to-end devices, providing flow control, error control, and segmentation.
   - o **Example**: TCP, UDP.
5. **Session Layer**:
   - o **Function**: Manages and controls the dialog (session) between two devices. It is responsible for establishing, maintaining, and terminating sessions.
   - o **Example**: NetBIOS, RPC.
6. **Presentation Layer**:
   - o **Function**: Translates, encrypts, and compresses data. It ensures that the data is in a readable format for the application.
   - o **Example**: JPEG, SSL/TLS.
7. **Application Layer**:
   - o **Function**: The closest layer to the end user. It provides services such as file transfer, email, and network management.
   - o **Example**: HTTP, FTP, SMTP.

2. Explain the TCP/IP model and compare it with the OSI model.

Answer: The **TCP/IP Model** is a simplified model for network communications and is used to define how data is transmitted across the internet. It has **4 layers**, as compared to the OSI model's 7 layers.

TCP/IP Model Layers:

1. **Application Layer**: Similar to the OSI Application, Presentation, and Session Layers. It handles end-user applications and services.
   - o Example: HTTP, FTP, DNS.
2. **Transport Layer**: Responsible for data transfer between systems. It corresponds to the OSI Transport layer.
   - o Example: TCP, UDP.
3. **Internet Layer**: Responsible for addressing, routing, and data packet forwarding, similar to the OSI Network layer.
   - o Example: IP (Internet Protocol), ARP (Address Resolution Protocol).
4. **Network Access Layer**: Corresponds to the OSI Data Link and Physical layers. It controls the physical hardware and network interface.
   - o Example: Ethernet, Wi-Fi.

TCP/IP vs OSI Diagram:
lua
Copy

```
 TCP/IP Model:          OSI Model:
 +-------------------+      +-------------------------+
 | Application Layer |      | Application Layer      |
 +-------------------+      +-------------------------+
```

```
| Transport Layer  |          | Presentation Layer    |
+------------------+          | Session Layer         |
| Internet Layer   |          +-----------------------+
+------------------+          | Transport Layer       |
| Network Access   |          +-----------------------+
+------------------+          | Network Layer         |
                    +-----------------------+
                    | Data Link Layer       |
                    +-----------------------+
                    | Physical Layer        |
                    +-----------------------+
```
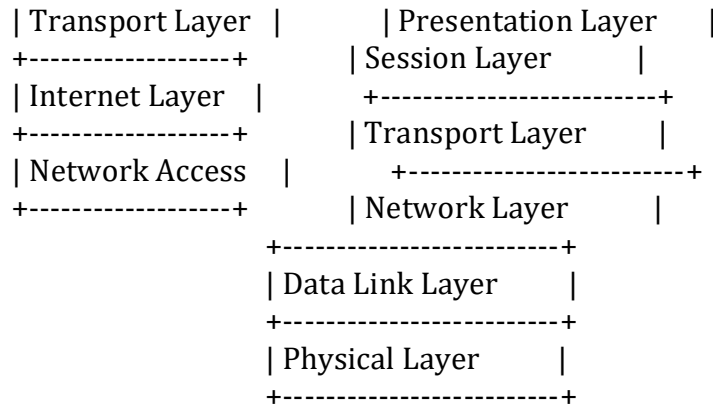
3. What is the concept of sockets in networking? Explain how a socket is created and used in a client-server model.

Answer: A **socket** is a software structure used for inter-process communication (IPC) over a network. It provides an interface between the application layer and the transport layer, allowing data exchange between clients and servers.

Client-Server Communication using Sockets:

1. **Server**:
   o Creates a socket with socket().
   o Binds the socket to a local address with bind().
   o Listens for incoming connections with listen().
   o Accepts a connection from a client with accept().
2. **Client**:
   o Creates a socket with socket().
   o Connects to the server's IP address and port using connect().
3. **Data Transfer**:
   o Both client and server exchange data using send() and recv() functions.
   o After communication, both close the socket using close().

Client-Server Sockets Diagram:
perl
Copy
```
    +------------------+          +------------------+
    |  Client Socket   |          |  Server Socket   |
    +------------------+          +------------------+
    | socket()       | <------------->| socket()        |
    | connect()      |          | bind()          |
    | send()/recv()   |<------------->| listen()        |
    | close()        |          | accept()        |
    +------------------+          | send()/recv()   |
                    | close()         |
```

```
                    +------------------+
```
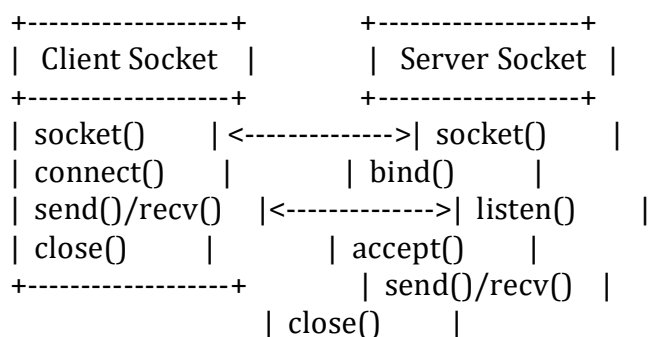
4. Explain the concept of DNS (Domain Name System). How does it work?

Answer: DNS (Domain Name System) is a hierarchical system that translates human-readable domain names (like www.example.com) into IP addresses used by network devices.

How DNS Works:

1. **DNS Query**: A user enters a domain name (e.g., www.example.com) in a web browser.
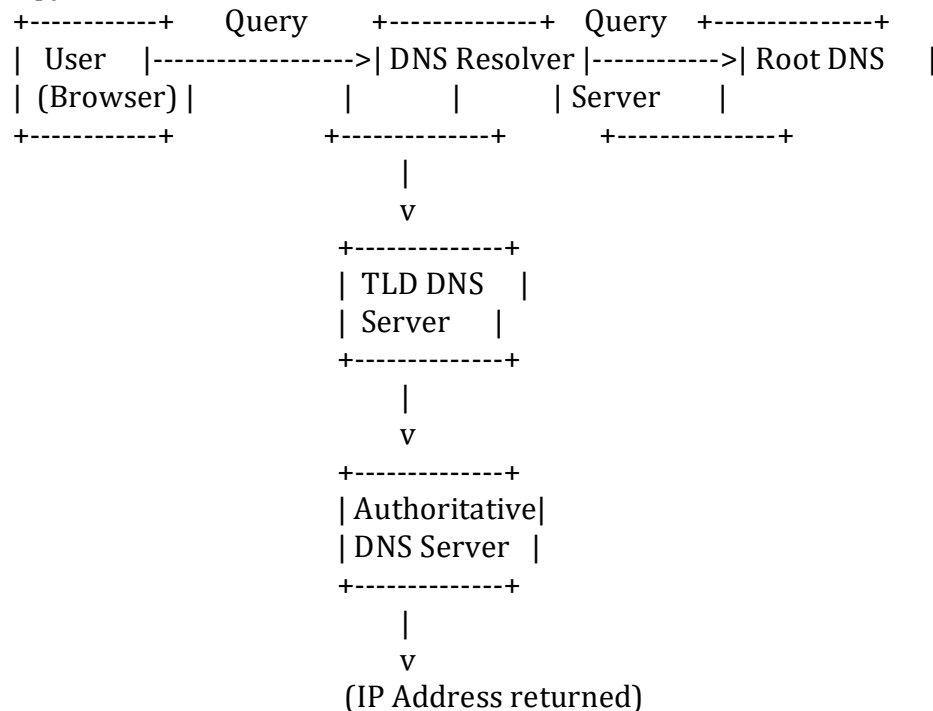2. **DNS Resolver**: The browser sends the query to a DNS resolver, which checks its local cache for the IP address.
3. **Root DNS Servers**: If the IP address isn't found, the resolver contacts the root DNS servers.
4. **TLD DNS Servers**: The resolver is directed to TLD DNS servers (like .com, .org).
5. **Authoritative DNS Server**: Finally, the resolver contacts the authoritative DNS server for the domain, which provides the IP address.
6. **Response**: The IP address is returned to the client, and the browser connects to the web server.

DNS Query Process Diagram:
pgsql
Copy
```
 +-----------+    Query    +--------------+  Query  +---------------+
 |  User     |------------->| DNS Resolver |-------->| Root DNS      |
 | (Browser) |             |              |         | Server        |
 +-----------+             +--------------+         +---------------+
                                  |
                                  v
                           +--------------+
                           | TLD DNS      |
                           | Server       |
                           +--------------+
                                  |
                                  v
                           +--------------+
                           | Authoritative|
                           | DNS Server   |
                           +--------------+
                                  |
                                  v
                            (IP Address returned)
```

5. Describe the FTP (File Transfer Protocol). How does it differ from HTTP?

Answer: FTP (File Transfer Protocol) is a protocol used for transferring files between client and server over a TCP/IP network. It supports both **uploading** and **downloading** files.

How FTP Works:

1. **Control Connection**: The client establishes a control connection to the FTP server on port 21.
2. **Data Transfer**: FTP uses a separate data connection (port 20) to transfer files.
3. **Modes**: FTP can operate in **active mode** or **passive mode** depending on how the data connection is established.

Differences between FTP and HTTP:

- **FTP** is used specifically for transferring files, while **HTTP** is used for retrieving web pages and resources.
- **FTP** allows file uploading, while **HTTP** does not.
- FTP uses two connections (control and data), while HTTP uses a single connection.

FTP vs HTTP Diagram:
lua
Copy
```
      FTP                  HTTP
 +-----------------+     +-----------------+
 | Control Channel |     | Single Channel  |
 | (Port 21)       |     | (Port 80)       |
 +-----------------+     +-----------------+
        |                     |
        v                     v
 +-----------------+     +-----------------+
 | Data Channel    |     | Web Content     |
 | (Port 20)       |     | (HTML, Images)  |
 +-----------------+     +-----------------+
```