Date :
P.No. :

Assignment - 4

1 What is the role of the Transport layer in the OSI model?

Ans → The transport layer in the OSI model is responsible for end-to-end communication between devices. It ensures reliable data transfer and provides services such as error detection, flow control, and segmentation/reassembly.

The role of the transport layer in the OSI model are-

1 Segmentation and Reassembly- Breaks down large messages into smaller segments and reassembles them at the destination.

2 End-to-End Communication- Establishes a logical connection between the sender and receiver.

3 Flow control - Manages data transmission speed to prevent overwhelming the receiver.

4 Error detection and Correction - Ensures data integrity using checksums and retransmissions.

5 Multiplexing and Demultiplexing - Allows multiple applications to use the same network connection.

Q What services does the Transport layer provide to applications?

A⇒ The transport layer in the OSI model provides several key services to applications, ensuring reliable and efficient communication between devices.
These services include

1 Process-to-Process Communication - It enables communication between specific applications running on different hosts using ports.

2 Multiplexing and Demultiplexing - It allows multiple applications to send and receive data over a

Date :
P.No. :

single network connection, distinguishing data using port numbers.

3 Reliable Data Transfer: Protocols like TCP ensure error-free, ordered delivery of data by using acknowledgements, retransmissions, and sequencing.

4 Flow control - It prevents a fast sender from overwhelming a slow receiver by adjusting the transmission rate (e.g., TCP uses a sliding window mechanism).

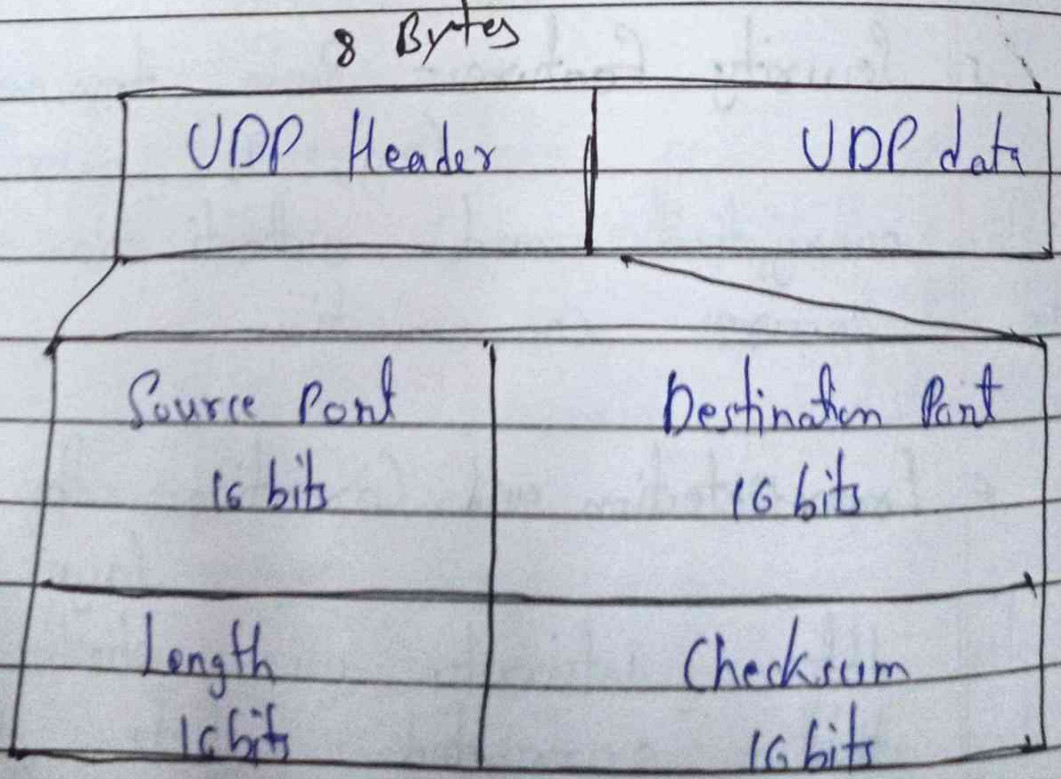
5 Security features - Some transport protocols (e.g., TLS over TCP) provide encryption and authentication to ensure secure communication.

6 Error Detection and Correction - The Transport layer verifies data integrity using checksums, ensuring that corrupted packets are detected and retransmitted.

Q. What is the UDP Header format?

Ans. UDP header has a fixed 8-bytes size simple header.

- The first 8 bytes contain all necessary header information and the remaining part consist of data.
- UDP port number fields are each 16 bits long, therefore the range for port number is defined from 0 to 65535. In that port number 0 is reserved.
- Port number help to differentiate several user requests or processes.



Date :
P.No. :

The UDP header contains four field are as follows

1 Source Port
- Source Port is a 16-bit field.
- It identifies the port of the sending application
- This field can be set to zero if the destination computer doesn't need to reply to the sender.

2 Destination Port
- Destination Port is a 16-bit field
- It identifies the port of the receiving application

3 Length
- Length is a 16-bit field.
- It identifies the combined length of UDP Header and Encapsulated data.

4 Checksum
- The checksum is a 16-bit field used for error control.
- It is calculated on UDP Header, encapsulated data, and IP pseudo-header.

P.No. :

4 What are the principles of reliable data transfer in the Transport layer?

Ans⇒ Transport layer Protocol are central piece of layered architectures, these provides the logical communication between application processes. These processes uses the logical communication to transfer data from transport layer to network layer and this transfer of data should be reliable and secure. The data is transferred in the form of packets but the problem occurs in reliable transfer of data.

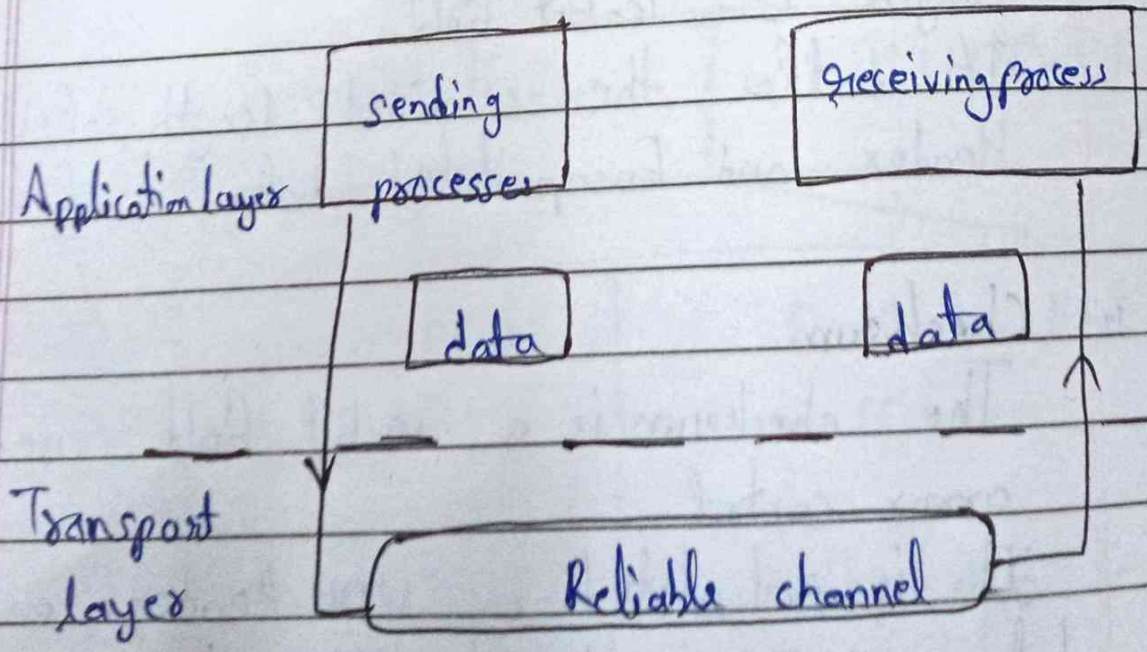


figure: Study of Reliable Data Transfer

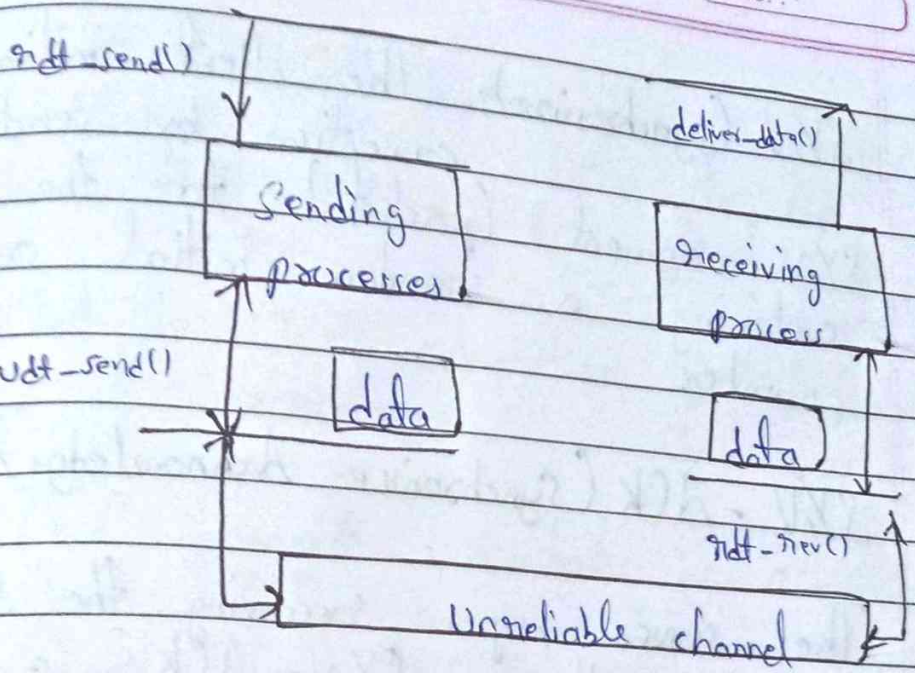


fig. Study of Unreliable Data Transfer

5 How does TCP establish a connection between two hosts?

A→ TCP establishes a connection between two hosts using a three-way handshake. The client sends a SYN, the server responds with SYN-ACK, and the client acknowledges with an ACK, ensuring both sides are synchronized and ready for data transmission.

Date :
P.No. :

SYN (Synchronize): The client initiates the connection by sending a SYN segment (packet) to the server, containing a random initial sequence number.

SYN - ACK (Synchronize- Acknowledge):

The server, upon receiving the SYN, responds with a SYN-ACK segment. This segment includes the server's own initial sequence no. and acknowledges the client's sequence number (by incrementing it by 1).

ACK (Acknowledge):

The client acknowledges the server's SYN-ACK by sending an ACK segment, acknowledging the server's sequence number (by incrementing it by 1).

Established Connection: Once these three segments are exchanged,

a reliable, full-duplex connection is established, allowing data to be transmitted between the client and server.

6 What is TCP congestion control, and why is it necessary?

A: TCP congestion control is a mechanism that TCP uses to manage data flow over a network and prevent congestion by adjusting the sending rate based on network conditions, ensuring efficient and reliable data transmission.

Why it is necessary

Preventing Network Overload: Without congestion control, multiple senders could flood the network with data, leading to packet loss, delays, and a general slowdown.

P.No. :

Ensuring Fair Resource Allocation - Congestion control helps ensure that all users and applications get a fair share of the network bandwidth, preventing a few users from monopolizing it.

Maintaining Reliable Communication: By adapting to network conditions and avoiding congestion, TCP congestion control helps ensure reliable and efficient data transmission.

7 What is SCTP, and how does it differ from TCP and UDP?

Ans⇒ SCTP is a relatively newer transport layer protocol that combines the advantage of both TCP and UDP. It offers the reliability of TCP while supporting message-oriented and real-time data transmission like UDP.

Ref:

Protocol	TCP	UDP	SCTP
1 Connection Type	Connection-oriented	Connectionless	Connection-oriented
2 Ordering	Guarantees ordered delivery of data packets	Does not guarantee ordered delivery of data packets	Guarantees ordered delivery of data packets
3 Multi homing	No	No	Yes
4 Multi-streaming	No	No	Yes
5 Speed	Slower due to reliability mechanisms	Faster due to minimal overhead	Comparable to TCP slower than UDP due to additional functionality
6 Error Recovery	Detects and retransmit lost or corrupted packets.	No error recovery mechanisms	Detects and retransmits lost or corrupted packets

Date :
P.No. :

Assignment-V

1. How does the Session layer differ from the Transport layer?

Ans:-

Parameter	Session layer	Transport layer
Function	Establishes, manages and terminates connections b/w two communicating devices or applications	Ensure reliable data delivery b/w end systems using protocols like TCP and UDP.
Focus:	Ensuring a stable and organized data exchange within a session.	Data segmentation, flow control and error correction to ensure data arrives correctly at the destination
Protocols:	Example include NetBIOS, SMB and RPC.	TCP and UDP are the primary protocol at this layer
Example	Maintaining a connection for a	TCP ensures reliable delivery

Date :
P.No. :

or ensuring a video stream remains uninterrupted while UDP allows for faster, but less reliable streaming of video.

Q2 What are the key functions of session management protocols?

Ans: Session management protocols handle the lifecycle of a user's interaction with a system, ensuring secure and efficient communication.

Key function include.

1. Authentication: Verifies user identity at the beginning of a session.

2. Authorization: Determines user permissions and access rights.

3 Session Establishment: Create and initializes a session, often with a unique session ID or token.

Date :
P.No. :

4 Session Maintenance - Keeps track of user interactions, managing timeouts and session state.

5 Session Recovery - Allows users to resume sessions if they are interrupted, often through token refresh mechanisms.

6 Session Termination - Properly ends a session after logout or inactivity to prevent unauthorized access.

Protocols like OAuth Connect, and Kerberos implement these functions to ensure secure and seamless user sessions.

Q3 What are common encryption techniques used in the Presentation layer

A⇒ The Presentation layer (layer 6) in the OSI model is responsible for data encryption, compression,

Date :
P.No. :

and translation. Common encryption techniques used at this layer include:

1. SSL / TLS (Secure Sockets Layer / Transport layer security)
   Encrypts data in transit to ensure confidentially and integrity.

2. AES (Advanced Encryption Standard)
   A widely used symmetric encryption algorithm (128-bit, 192-bit and 256 bit keys).

   Provides fast and secure encryption for transmitted data.

3. RSA (Rivest-Shamir-Adleman)

   - A public-key cryptosystem used in key exchange and digital signatures.
   - Often used in SSL/TLS handshakes to establish secure connections.

Date :
P.No. :

4 SHA (Secure Hash Algorithms)

- SHA-2 and SHA-3 provide cryptographic hashing for data integrity verification.
- Often used in conjunction with TLS and digital signatures.

Q4 What is DNS, and why is it important for the Application layer?

Ans DNS (Domain Name System) is a distributed database system that translates human-friendly domain names (like "example.com") into the numerical IP addresses (like "93.184.216.34") that computers use to communicate, and it's crucial for the Application layer because it allows applications to use these names instead of IP addresses.

DNS is important for the Application layer.

Date :
P.No. :

Use-friendliness:- Applications, like web browsers, rely on DNS to translate domain names into IP addresses, making it easier for users to navigate the internet.

Application functionality: Applications need to know the IP address of the server they want to communicate with, and DNS provides that information.

Ubiquity: DNS is a fundamental part of the internet infrastructure, used by virtually all applications that access the web.

5 What is SNMP, and what role does it play in network management?

Ans: SNMP (Simple Network Management Protocol) is a widely used protocol

for monitoring and managing devices on IP networks, including routers, switches, servers, printers, and other network hardware. It allows network administrators to collect performance data, detect faults, and configure devices remotely.

Role of SNMP in Network Management

Monitoring Network Performance - SNMP collects data on metrics such as bandwidth usage, CPU load, and memory utilization, helping administrators optimize network efficiency.

Fault Detection & Alerts - Devices can send SNMP traps (alerts) when specific events occur, such as hardware failures or high traffic loads, enabling quick response to issues.

Remote Configuration & Management - Network administrators

can use SNMP to modify device settings, such as changing IP addresses, restarting devices, or updating configurations.

Security & Access Control - SNMP can be configured to restrict access to management data, ensuring only authorized users can view or modify network settings.

5 What is the differences between SMTP, POP3 and IMAP?

Purpose	SMTP	POP3	IMAP
1. Full form	Simple Mail Transfer Protocol.	Post office Protocol 3	Internet Message Access Protocol.
2. Purpose	Used for sending emails from our device	Used for retrieving email from the server to our device	Used for retrieving emails from the server, allowing access from multiple devices

Function	Establishes how the message gets from the sender to the mail server.	Downloads all emails from the server to our device and, by default, deletes them from the server.	Keeps emails on the server, allowing you to access them from any device with an internet connection.
Advantages	When we compose and send an email, your email client uses SMTP to transmit the messages to the recipient's email server.	. Offline access to emails . Reduced server storage space usage. . Fast attachment viewing.	. Synchronization of emails across multiple devices. . Access to emails from any device. . Organized email management on the server.

Date :
P.No. :

7 What are the different types of HTTP method (GET, POST, PUT, DELETE)?

Ans→ HTTP methods (also known as HTTP verbs) define the actions that can be performed on a resource in a RESTful API. The most commonly used HTTP methods are:

1. GET - Used to retrieve data from a server.
   e.g. Fetching a list of users (GET /users).
   It is safe (does not modify data) and idempotent (multiple requests yield the same result).

2. POST - Used to send data to a server to create a new resource.
   e.g. Creating a new user (POST /users) with user data in the request body).
   - It is not idempotent (multiple requests

body).

It is not idempotent (multiple requests can create multiple records).

3. PUT - Used to update an existing resource, or create one if it does not exist.

e.g - Updating user details (PUT /users/123 with updated data).

It is idempotent (sending the same request multiple times results in the same outcome.

4. DELETE - Used to remove a resource from the server.

e.g. deleting a user (DELETE /users/123).

It is idempotent (deleting the same resource multiple times has the same effect.