# Transport Layer

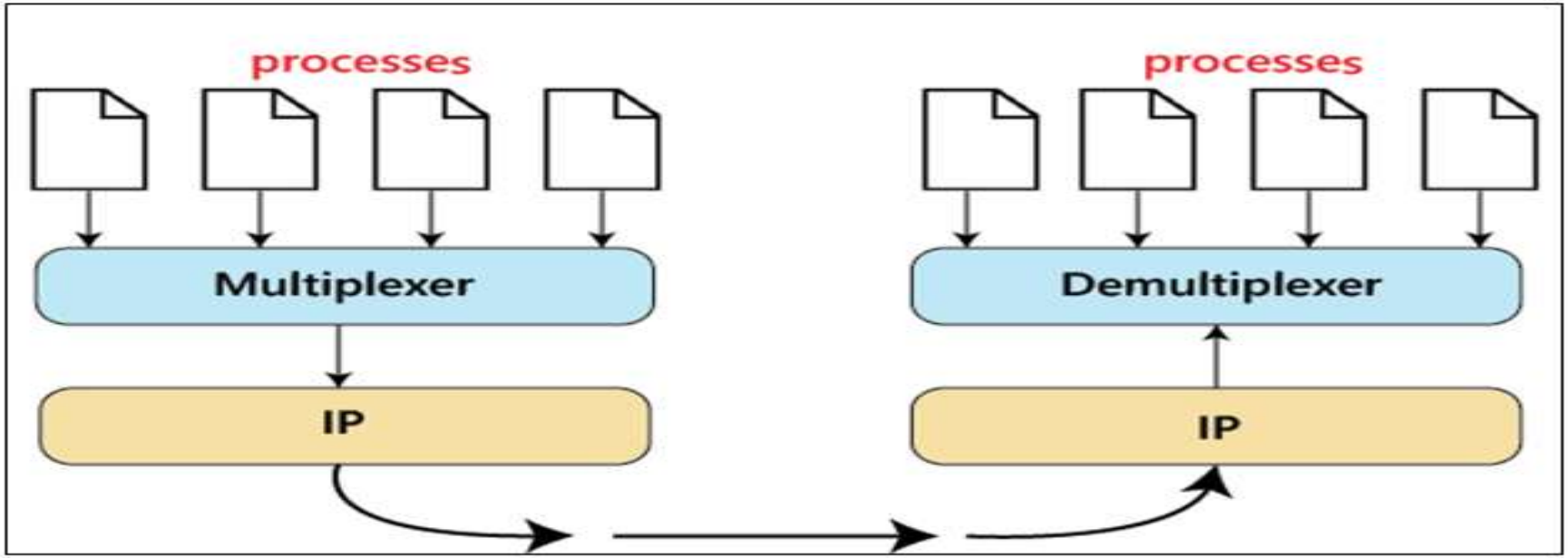
PANA ACADEMY

# Services of Transport Layer

## Responsibilities of a Transport Layer

- The Process to Process Delivery
- End-to-End Connection between Hosts
- Multiplexing and Demultiplexing
- Congestion Control
- Data integrity and Error correction
- Flow control

# Diagram Explaining Process to Process Communication




*Process to Process Delivery*

# TCP VS UDP

| TCP | UDP |
| --- | --- |
| TCP is a connection-oriented protocol. | UDP is the Connection less protocol. |
| TCP is reliable as it guarantees the delivery of data to the destination router. | The delivery of data to the destination cannot be guaranteed in UDP. |
| An acknowledgment segment is present. | No acknowledgment segment. |
| Sequencing of data is a done in this protocol. | There is no sequencing of data in UDP. |
| TCP is comparatively slower than UDP. | UDP is faster, simpler, and more efficient than TCP. |
| TCP has a (20-60) bytes variable length header. | UDP has an 8 bytes fixed-length header. |
| Uses handshakes such as SYN, ACK, SYN-ACK | It's a connectionless protocol i.e. No handshake |
| TCP is used by HTTP, HTTPs, FTP , SMTP and Telnet. | UDP is used by DNS, DHCP, TFTP,SNMP, RIP, and VoIP. |

# Port and Socket

A port is a logical construct assigned to network processes so that they can be identified within the system.

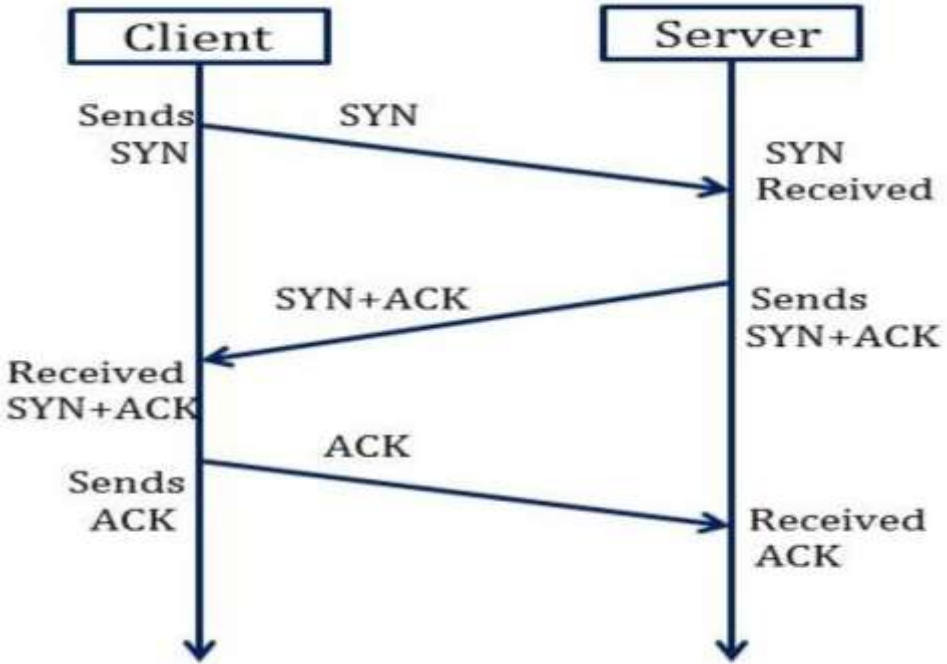
A socket is a combination of port and IP address. (IP ADDRESS + PORT)

There are 65,535 possible port numbers, although not all are in common use. Some of the most commonly used ports, along with their associated networking protocol, are

0 - 1023 are well known ports

| Service, Protocol, or Application | Port Number | TCP or UDP |
|---|---|---|
| FTP (File Transfer Protocol) | 20, 21 | TCP |
| SSH (Secure Shell Protocol) | 22 | TCP |
| Telnet | 23 | TCP |
| SMTP (Simple Mail Transfer Protocol) | 25 | TCP |
| DNS (Domain Name System | 53 | UDP |
| TFTP | 69 | UDP |
| HTTP | 80 | TCP |
| POP3 | 110 | TCP |
| IMAP4 | 143 | TCP |

# TCP Three-way Handshake



# Explanation of Three-way handshake Mechanism

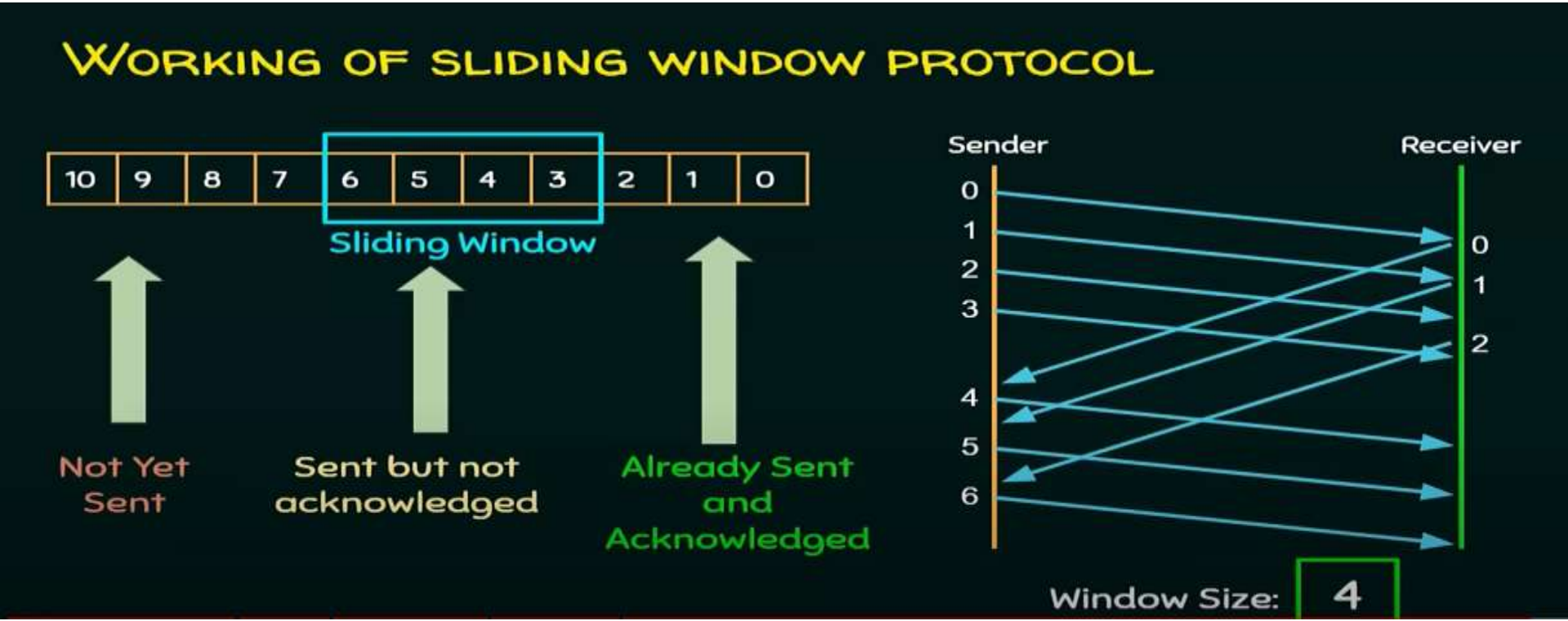
1. Connection Initiation:
    - Client sends a SYN (synchronize) packet to the server, indicating its intent to establish a connection.
    - The SYN packet contains a sequence number chosen by the client to start the connection.
2. Acknowledgment and Agreement:
    - Upon receiving the SYN packet, the server responds with a SYN-ACK (synchronize-acknowledgment) packet.
    - The SYN-ACK packet acknowledges the client's SYN packet and contains the server's own chosen sequence number.
3. Finalizing the Connection:
    - Finally, the client acknowledges the server's SYN-ACK packet by sending an ACK packet.
    - This ACK packet confirms the server's acknowledgment and completes the three-way handshake.

# Flow Control In Transport Layer

Flow control mechanisms in the transport layer regulate the rate of data transmission between sender and receiver to ensure that the sender does not overwhelm the receiver.

1. Sliding Window Protocol:
    - TCP uses a sliding window protocol for flow control. This window represents the amount of data that the sender can transmit before receiving an acknowledgment from the receiver.
    - Send Multiple Frames at a time.
    - Number of frames to be sent is based on window size.
    - Each frame is numbered which we call as sequence number.

# Working of Sliding Window



# Congestion Control

A state occurring in network layer when the message traffic is so heavy that it slows down network response time.

Congestion Control is a mechanism that controls the entry of data packets into the network, enabling a better use of a shared network infrastructure and avoiding congestive collapse.

# Comparison of Leaky Bucket and Token Bucket

- Both algorithms are used for traffic shaping and policing, but they have different approaches to controlling the rate of data transmission.
- The leaky bucket algorithm maintains a constant output rate, while the token bucket algorithm allows for bursts of traffic as long as tokens are available.
- The token bucket algorithm offers more flexibility in regulating traffic and supporting different QoS levels compared to the leaky bucket algorithm.