Unit-03

Transport Layer

3.1. Introduction and Transport-Layer Services

Transport Layer

The transport layer is the fourth layer in the OSI (Open Systems Interconnection) model and plays a crucial role in the architecture of network communication. It serves as an intermediary between the application layer (which provides user-facing services) and the network layer (which handles data routing). The primary purpose of the transport layer is to provide reliable data transfer, manage end-to-end communication, and ensure that data is transmitted accurately and in the correct order.

Transport-Layer Services

The transport layer in the OSI (Open Systems Interconnection) model provides crucial services that facilitate end-to-end communication between devices across a network. These services are essential for managing data transmission, ensuring reliability, and maintaining the overall efficiency and integrity of network communications.

Transport-Layer Services

1. **Reliable Data Transfer**:

   Ensures accurate and ordered delivery of data using error detection, acknowledgments, and retransmissions.

2. **Connection Establishment and Termination**:

   Manages the start and end of communication sessions, synchronizing sender and receiver.

3. **Flow Control**:

   Regulates data transmission rate to prevent sender from overwhelming the receiver.

4. **Multiplexing and Demultiplexing**:

Uses port numbers to manage multiple data streams from different applications.

5. **Congestion Control**:

   Adjusts data transmission rate to prevent network congestion and ensure fair bandwidth usage.

6. **Error Detection and Correction**:

   Identifies and corrects errors in transmitted data using checksums and other mechanisms.

7. **Data Segmentation and Reassembly**:

   Breaks data into smaller segments for transmission and reassembles them at the destination.

8. **Quality of Service (QoS)**:

   Manages bandwidth and prioritizes traffic for applications requiring specific performance levels.

3.1.1. Relationship Between Transport and Network Layers

1. Data Encapsulation:

Transport layer encapsulates data into segments; network layer encapsulates segments into packets with IP addresses.

2. Addressing:

Network layer handles IP addresses for routing; transport layer uses port numbers to direct data to the correct application.

3. End-to-End Communication:

Transport layer ensures reliable data transfer between devices; network layer manages packet delivery across networks.

4. Error Handling and Reliability:

Transport layer manages error detection, correction, and retransmissions; network layer does not guarantee reliable delivery.

5. Flow Control and Congestion Management:

Transport layer manages flow control between sender and receiver; network layer handles congestion management to prevent network overload.

6. Quality of Service (QoS):

Both layers contribute to QoS by prioritizing data types and ensuring efficient packet handling and routing.

3.1.2. Overview of the Transport Layer in the Internet

The transport layer is a crucial component of the Internet's network architecture, responsible for providing end-to-end communication services between applications running on different hosts. It is the fourth layer in the OSI model and corresponds to the transport layer in the TCP/IP model. This layer ensures that data is transmitted reliably and efficiently, despite potential issues in the underlying network.

3.2. Multiplexing and De-multiplexing

Multiplexing and demultiplexing are key functions of the transport layer in network communication. These processes enable multiple applications or services to share the same network connection and ensure that the data sent and received is directed to the appropriate application.

Multiplexing

Multiplexing is the process of combining data from multiple applications or services for transmission over a single network channel. The transport layer at the sender's side performs this function. It involves assigning unique identifiers, called port numbers, to each data stream. These port numbers, combined with the IP address, create a unique identifier for each communication session.

Demultiplexing

Demultiplexing is the process of receiving data and directing it to the appropriate application based on the associated port number. This process occurs at the receiving end of a communication session. The transport layer examines the port number in the incoming segments or datagrams and uses this information to forward the data to the correct application.

Importance of Multiplexing and Demultiplexing

- Efficient Use of Resources
- Simplified Communication
- Flexibility

3.3. Connectionless Transport

Connectionless transport is a communication method in networking where data is sent between devices without establishing a dedicated connection. In this model, each data packet is treated independently, and there is no session or state information maintained between the sender and receiver. The most common protocol that uses connectionless transport is the User Datagram Protocol (UDP).

3.3.1. UDP Segment Structure

User Datagram Protocol (UDP)

User Datagram Protocol (UDP) is a connectionless transport layer protocol used in networking. It is defined in the Internet Protocol Suite, alongside other protocols such as TCP (Transmission Control Protocol). UDP is known for its simplicity and minimal overhead, making it suitable for applications where speed and efficiency are prioritized over reliability.

3.3.2. UDP Checksum

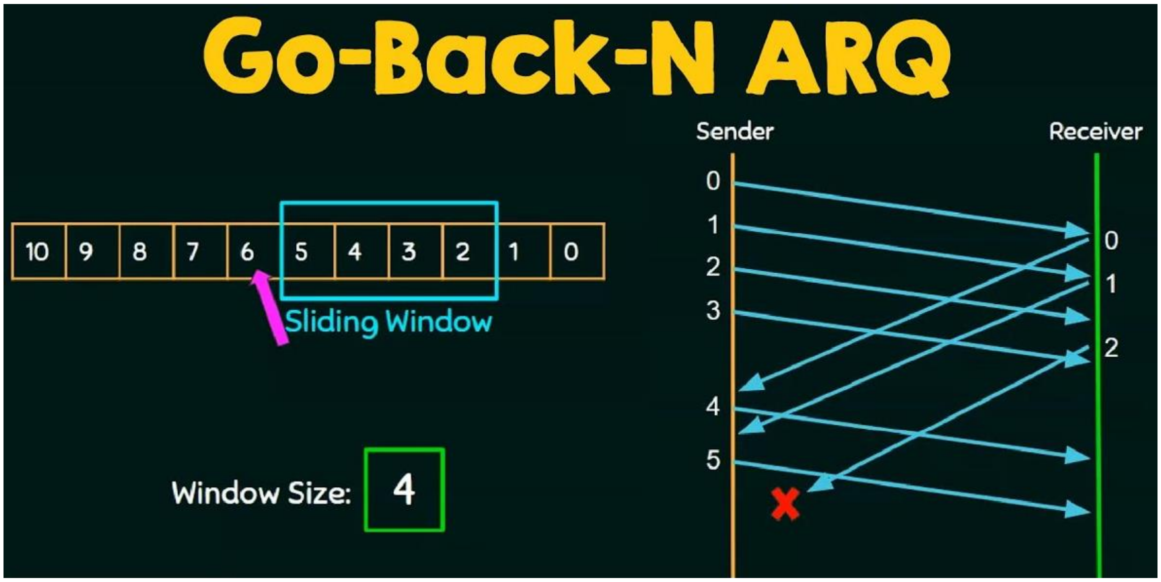
he UDP Checksum is an optional 16-bit field in the User Datagram Protocol (UDP) header. Its primary purpose is to provide basic error-checking for the data integrity of the UDP datagram. The checksum helps detect errors that may occur during transmission, such as bit errors caused by noise in the network.

3.4. Principles of Reliable Data Transfer

The principles of reliable data transfer are essential for ensuring accurate and ordered delivery of data over a network. These principles include error detection, positive and negative acknowledgments, retransmission, sequence numbering, timers and timeouts, flow control, and congestion control. They are implemented in protocols like TCP to provide reliable communication, essential for many applications and services on the Internet.

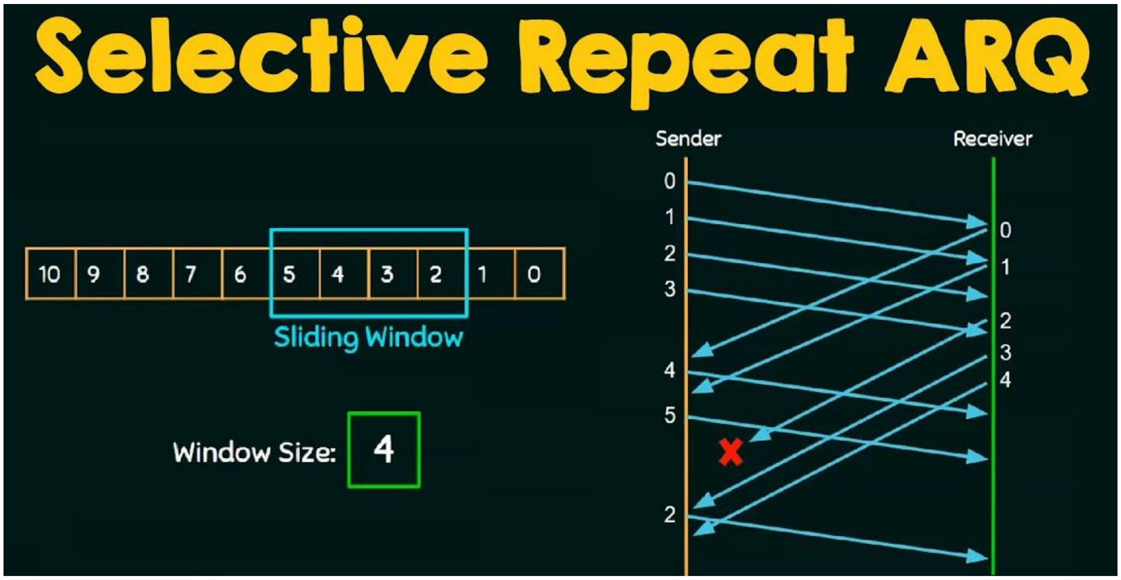
3.4.1. Go-Back-N (GBN)

Go-Back-N (GBN) is an Automatic Repeat reQuest (ARQ) protocol used in networking to provide reliable data transfer over an unreliable communication channel. It is a type of sliding window protocol that ensures data integrity and order through acknowledgments and retransmissions.



3.4.2. Selective Repeat (SR)

Selective Repeat (SR) is an Automatic Repeat reQuest (ARQ) protocol that provides reliable data transfer by allowing the sender and receiver to handle data more efficiently compared to Go-Back-N (GBN). SR improves efficiency by retransmitting only the specific frames that are lost or contain errors, rather than all subsequent frames.



3.5. Connection-Oriented Transport

The Transmission Control Protocol (TCP) is a connection-oriented transport layer protocol in the Internet Protocol (IP) suite. It is designed to provide reliable, ordered, and error-checked delivery of data between applications running on hosts in a network. TCP is widely used for applications that require reliable communication, such as web browsing, email, and file transfers.

3.5.1. Round-Trip Time Estimation and Timeout

❖ **Round-Trip Time (RTT)**

→ Round-Trip Time (RTT) is the duration it takes for a signal or data packet to travel from a source to a destination and then back to the source.

❖ **Round-Trip Time Estimation**

→ Round-Trip Time (RTT) estimation involves predicting the time it takes for a signal or data packet to travel from a source to a destination and back.

❖ **Round-Trip Time Timeout (RTT Timeout)**

→ It defines the maximum time a sender will wait for an acknowledgment (ACK) of a transmitted packet before assuming that the packet was lost or not received.

3.5.2. Reliable Data Transfer

Reliable Data Transfer (RDT) refers to the process of ensuring that data is transmitted accurately and completely from a sender to a receiver over a network, despite potential issues like packet loss, errors, or network congestion. This concept is fundamental in network communication protocols, such as TCP (Transmission Control Protocol), which guarantee reliable delivery of data.

3.5.3. Flow Control

Flow control is a mechanism used in data communication to manage the rate at which data is transmitted between a sender and a receiver. Its primary purpose is to ensure that the sender does not overwhelm the receiver with too much data too quickly, which could lead to data loss, network congestion, or inefficient use of resources.

3.6. TCP Congestion Control

TCP Congestion Control is a set of algorithms and mechanisms implemented in the Transmission Control Protocol (TCP) to manage network congestion, ensuring efficient data transmission while preventing network overload. It adjusts the rate at which data is sent based on current network conditions, balancing the need for high throughput with the goal of minimizing packet loss and delays.