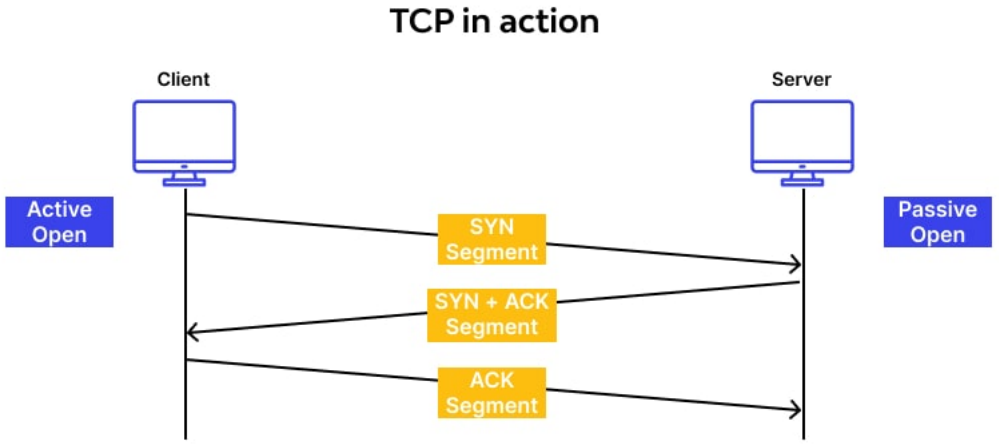
Chapter 4

TCP/UDP : Transport layer protocols

Master 1 IS , Guelma University, 2024-2025

Introduction to TCP

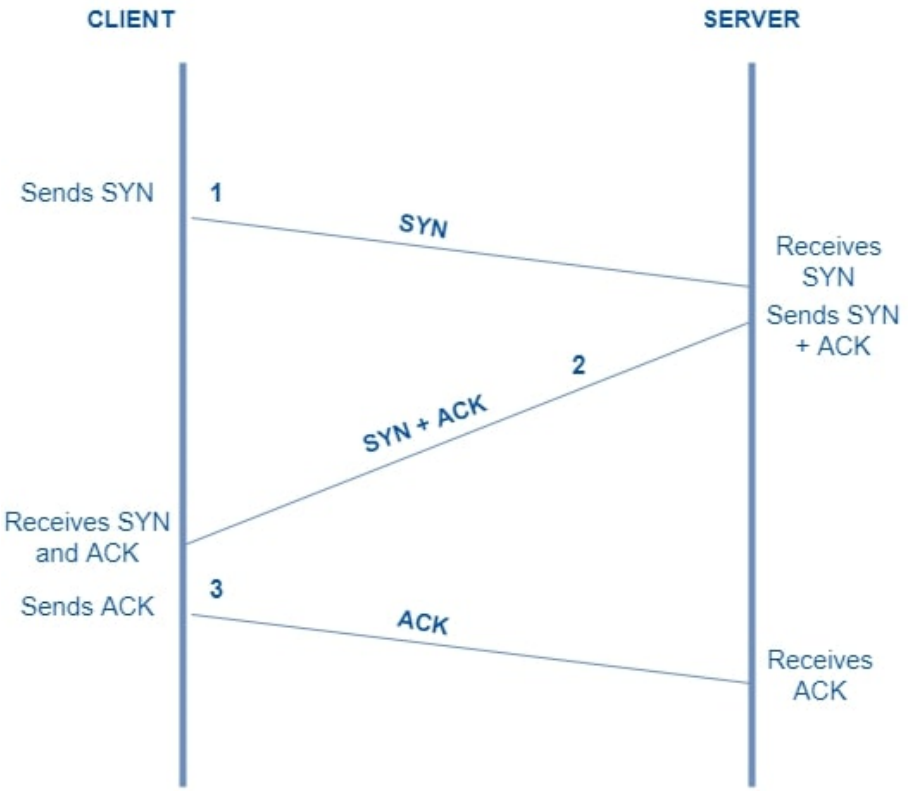
- **Transmission Control Protocol (TCP)** is a **connection-oriented protocol for communications** that helps in the exchange of messages between different devices over a network. It is one of the main protocols of the TCP/IP suite. In OSI model, it operates at the transport layer(Layer 4). It lies between the Application and Network Layers which are used in providing reliable delivery services. The Internet Protocol (IP), which establishes the technique for sending data packets between computers, works with TCP.
- TCP establishes a reliable connection between sender and receiver using the **three-way handshake (SYN, SYN-ACK, ACK)** and it uses a **four-step handshake (FIN, ACK, FIN, ACK)** to close connections properly.
- It ensures **error-free, in-order delivery** of data packets.
- It uses **acknowledgments (ACKs)** to confirm receipt.
- It prevents data overflow by adjusting the data transmission rate according to the receiver's buffer size.
- It prevents network congestion using algorithms like **Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery.**
- TCP header uses **checksum** to detect corrupted data and requests retransmission if needed.
- It is used in applications requiring **reliable** and **ordered** data transfer, such as web browsing, email, and remote login.



No modifications since their development around 30 years ago…

TCP 3-Way Handshake Process

- The TCP 3-Way Handshake is a fundamental process that establishes a reliable connection between two devices over a TCP/IP network. It involves three steps: SYN (Synchronize), SYN-ACK (Synchronize-Acknowledge), and ACK (Acknowledge).
- During the handshake, the client and server exchange initial sequence numbers and confirm the connection establishment.
- **Step 1 (SYN):** In the first step, the client wants to establish a connection with a server, so it sends a segment with SYN(Synchronize Sequence Number) which informs the server that the client is likely to start communication and with what sequence number it starts segments with
- **Step 2 (SYN + ACK):** Server responds to the client request with SYN-ACK signal bits set. Acknowledgement(ACK) signifies the response of the segment it received and SYN signifies with what sequence number it is likely to start the segments with
- **Step 3 (ACK):** In the final part client acknowledges the response of the server and they both establish a reliable connection with which they will start the actual data transfer



TCP Segment Structure

A TCP segment consists of data bytes to be sent and a header that is added to the data by TCP as shown:

- **Size:** 20-60 bytes. No options: 20 bytes; with options: up to 60 bytes.
- **Source Port & Destination Port:** 16-bit fields for sending/receiving application ports.
- **Sequence Number:** 32 bits; indicates the first byte's sequence in the segment.
- **Acknowledgement Number:** 32 bits; indicates the next expected byte.
- **Header Length (HLEN):** 4 bits; represents header size in 4-byte words (range: 5–15).
- **Control Flags (6 bits):**

URG: Urgent pointer valid

ACK: Acknowledgement valid

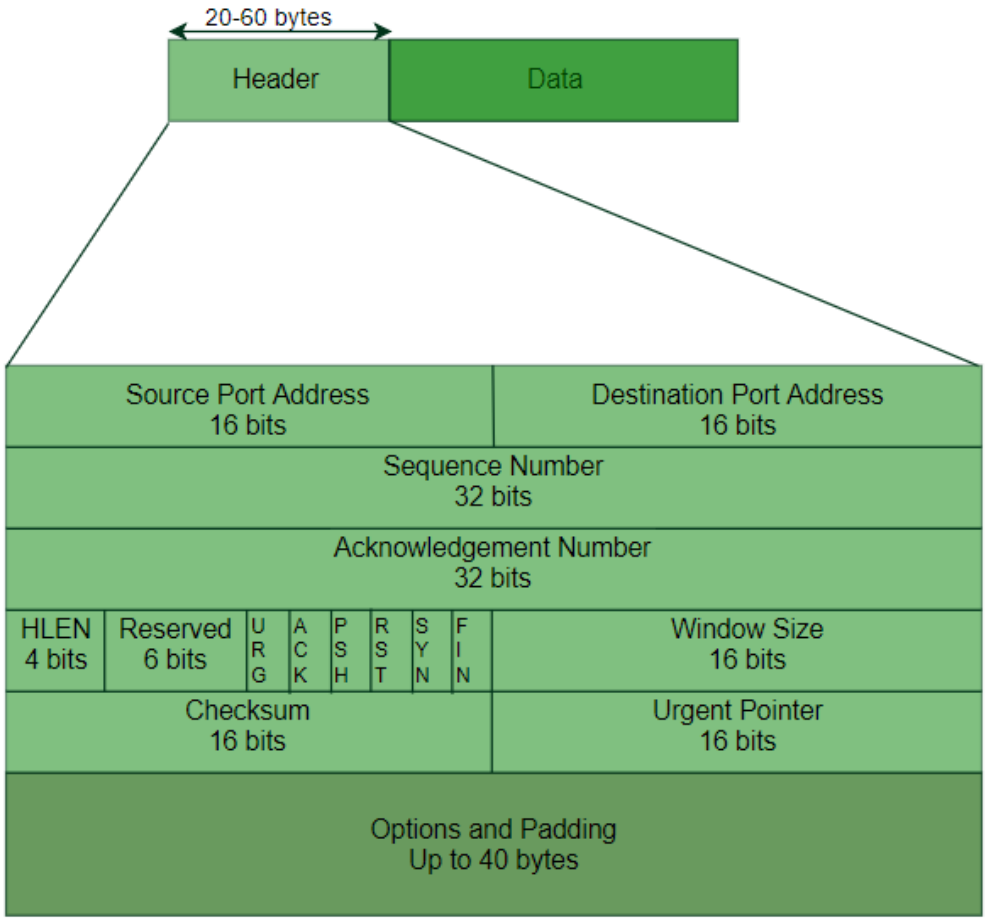
PSH: Request push

RST: Reset connection

SYN: Synchronize sequence numbers

FIN: Terminate connection

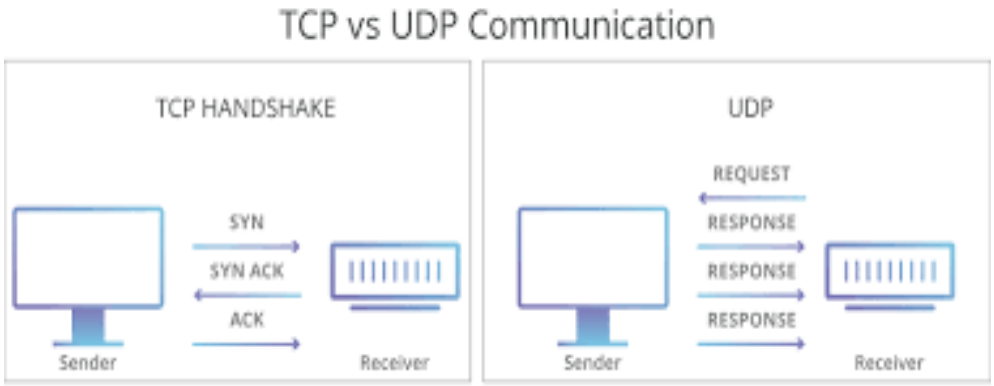
- **Window Size:** Indicates the sending TCP's window size in bytes.
- **Checksum:** 16-bit field for error control.
- **Urgent Pointer:** Points to the last urgent byte (active if URG flag is set).



User Datagram Protocol (UDP)

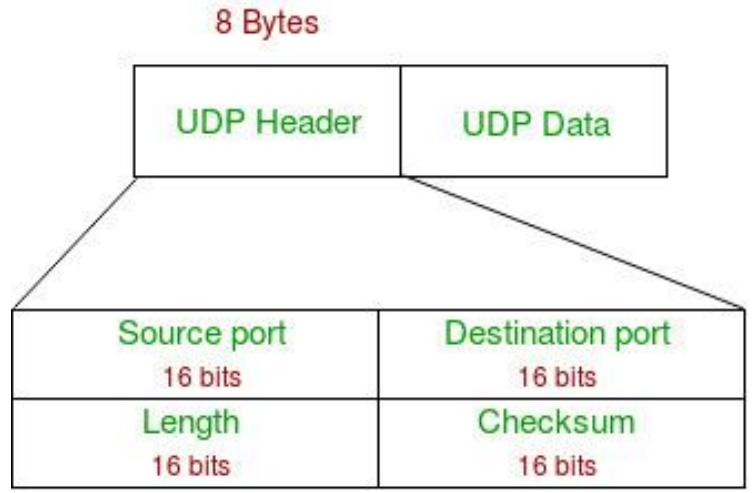
What is User Datagram Protocol?

- User Datagram Protocol (UDP) is one of the core protocols of the Internet Protocol (IP) suite. It is a communication protocol used across the internet for time-sensitive transmissions such as video playback or DNS lookups . Unlike Transmission Control Protocol (TCP), UDP is connectionless and does not guarantee delivery, order, or error checking, making it a lightweight and efficient option for certain types of data transmission.



UDP Header

- UDP header is an **8-byte** fixed and simple header, while for TCP it may vary from 20 bytes to 60 bytes. The first 8 Bytes contain all necessary header information and the remaining part consists of data. UDP port number fields are each 16 bits long, therefore the range for port numbers is defined from 0 to 65535; port number 0 is reserved. Port numbers help to distinguish different user requests or processes.
- **Source Port:** Source Port is a 2 Byte long field used to identify the port number of the source.
- **Destination Port:** It is a 2 Byte long field, used to identify the port of the destined packet.
- **Length:** Length is the length of UDP including the header and the data. It is a 16-bits field.
- **Checksum:** Checksum is 2 Bytes long field. It is the 16-bit one's complement of the one's complement sum of the UDP header, the pseudo-header of information from the IP header, and the data, padded with zero octets at the end (if necessary) to make a multiple of two octets.



Notes – Unlike TCP, the Checksum calculation is not mandatory in UDP. No Error control or flow control is provided by UDP. Hence UDP depends on IP and ICMP for error reporting. Also UDP provides port numbers so that is can differentiate between users requests.

Applications of UDP

- Used for simple request-response communication when the size of data is less and hence there is lesser concern about flow and error control.
- It is a suitable protocol for multicasting as UDP supports packet switching.
- UDP is used for some routing update protocols like RIP(Routing Information Protocol).
- Normally used for real-time applications which can not tolerate uneven delays between sections of a received message.
- VoIP (Voice over Internet Protocol) services, such as Skype and WhatsApp, use UDP for real-time voice communication. The delay in voice communication can be noticeable if packets are delayed due to congestion control, so UDP is used to ensure fast and efficient data transmission.
- DNS (Domain Name System) also uses UDP for its query/response messages. DNS queries are typically small and require a quick response time, making UDP a suitable protocol for this application.
- DHCP (Dynamic Host Configuration Protocol) uses UDP to dynamically assign IP addresses to devices on a network. DHCP messages are typically small, and the delay caused by packet loss or retransmission is generally not critical for this application.

TCP vs UDP

Basis	Transmission Control Protocol (TCP)	User Datagram Protocol (UDP)
Type of Service	TCP is a connection-oriented protocol. Connection orientation means that the communicating devices should establish a connection before transmitting data and should close the connection after transmitting the data.	UDP is the Datagram-oriented protocol. This is because there is no overhead for opening a connection, maintaining a connection, or terminating a connection. UDP is efficient for broadcast and multicast types of network transmission.
Reliability	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.
Error checking mechanism	TCP provides extensive error-checking mechanisms. It is because it provides flow control and acknowledgment of data.	UDP has only the basic error-checking mechanism using checksums.
Acknowledgment	An acknowledgment segment is present.	No acknowledgment segment.
Sequence	Sequencing of data is a feature of Transmission Control Protocol (TCP). this means that packets arrive in order at the receiver.	There is no sequencing of data in UDP. If the order is required, it has to be managed by the application layer.
Speed	TCP is comparatively slower than UDP.	UDP is faster, simpler, and more efficient than TCP.
Retransmission	Retransmission of lost packets is possible in TCP, but not in UDP.	There is no retransmission of lost packets in the User Datagram Protocol (UDP).
Header Length	TCP has a (20-60) bytes variable length header.	UDP has an 8 bytes fixed-length header.
Weight	TCP is heavy-weight.	UDP is lightweight.
Handshaking Techniques	Uses handshakes such as SYN, ACK, SYN-ACK	It's a connectionless protocol i.e. No handshake
Broadcasting	TCP doesn't support Broadcasting.	UDP supports Broadcasting.
Protocols	TCP is used by HTTP, HTTPs , FTP , SMTP and Telnet .	UDP is used by DNS, DHCP, TFTP, SNMP , RIP, and VoIP.
Stream Type	The TCP connection is a byte stream.	UDP connection is a message stream.
Overhead	Low but higher than UDP.	Very low.
Applications	This protocol is primarily utilized in situations when a safe and trustworthy communication procedure is necessary, such as in email, on the web surfing, and in military services.	This protocol is used in situations where quick communication is necessary but where dependability is not a concern, such as VoIP, game streaming, video, and music streaming, etc.

How is UDP used in DDoS attacks?

- A **UDP flood attack** is a type of **Distributed Denial of Service (DDoS)** attack where an attacker sends a large number of **User Datagram Protocol (UDP)** packets to a target port.
- **UDP Protocol** : Unlike TCP, UDP is connectionless and doesn't require a handshake before data transfer. When a UDP packet arrives at a server, it checks the specified port for listening applications. If no app is found, the server sends an **ICMP "destination unreachable"** packet to the supposed sender (usually a random bystander due to spoofed IP addresses).
- **Attack Process** :
  - The attacker sends UDP packets with spoofed IP sender addresses to random ports on the target system.
  - The server checks each incoming packet's port for a listening application (usually not found due to random port selection).
  - The server sends ICMP "destination unreachable" packets to the spoofed sender (random bystanders).
  - The attacker floods the victim with UDP data packets, overwhelming its resources.