Internet Architecture and Protocols

**Submission date: 17 May 2023**

**Submitted to:**

Miss Quratulain

***Assignment Topic:***

**TCP and UDP**

University of

Education

dldldld

**Submitted by**

Adeeb Hassan

Roll #:bsf2003459

Department: BS IT

(Morning,2020-24)

)

**Transmission Control Protocol(TCP) and User Datagram Protocol(UDP)**

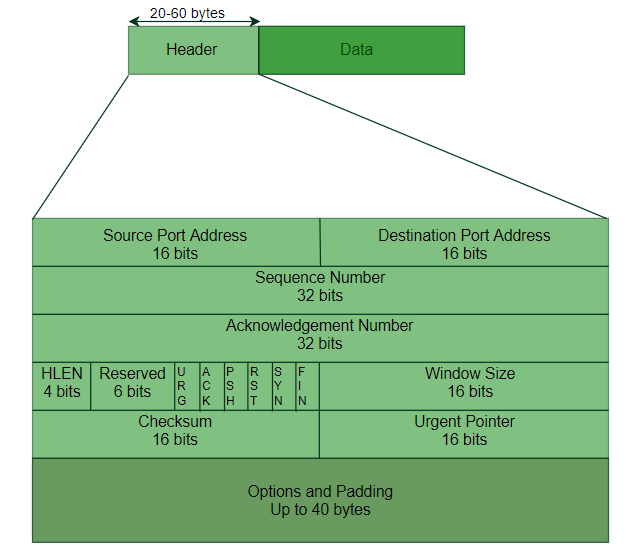
**Transmission Control Protocol (TCP):**

TCP is a communications standard that enables application programs and computing devices to exchange messages over a network. It is designed to send [packets](https://www.fortinet.com/resources/cyberglossary/what-is-packet-loss) across the internet and ensure the successful delivery of data and messages over networks.

**TCP used for:**

TCP enables data to be transferred between applications and devices on a network and is used in the TCP IP model. It is designed to break down a message, such as an email, into packets of data to ensure the message reaches its destination successfully and as quickly as possible.

**TCP Header:**

****

**Source Port (16 bits):** This field identifies the port number of the sender's application or process.

**Destination Port (16 bits):** This field indicates the port number of the receiver's application or process.

**Sequence Number (32 bits):** The sequence number field assigns a unique number to each byte of data in the TCP segment. It helps in ordering and reassembling received segments at the receiver's end.

**Acknowledgment Number (32 bits):** In an ACK (acknowledgment) packet, this field indicates the next expected sequence number the receiver is anticipating. It acknowledges the successful reception of all the bytes up to the acknowledged number.

**Data Offset (4 bits):** The data offset field specifies the length of the TCP header in 32-bit words. It indicates the start of the data section following the TCP header.

**Reserved (6 bits):** These bits are currently reserved for future use and should be set to zero.

**Control Flags (6 bits):** This field contains several control flags, also known as TCP flags or control bits. These flags include:

* *URG (Urgent):* It indicates that the Urgent Pointer field is significant and should be examined.
* *ACK (Acknowledgment):* It signifies that the Acknowledgment Number field is valid.
* *PSH (Push):* It requests immediate data delivery to the receiving application.
* *RST (Reset):* It resets the connection or indicates an error condition.
* *SYN (Synchronize):* It initiates a connection establishment between sender and receiver.
* *FIN (Finish):* It signifies the end of data transmission and initiates the connection termination process.

**Window Size (16 bits):** This field specifies the size of the receiving window. It indicates the number of bytes the sender can transmit before requiring an acknowledgment.

**Checksum (16 bits):** The checksum field contains a value computed over the TCP header, data, and a pseudo-header. It ensures the integrity of the TCP segment by detecting transmission errors.

**Urgent Pointer (16 bits):** If the URG flag is set, this field points to the last byte of urgent data in the TCP segment.

**Options (variable length):** The options field is optional and used for additional features and extensions in TCP. It may include options such as Maximum Segment Size (MSS), Selective Acknowledgment (SACK), and Timestamps.

**Padding (variable length):** If the options field is present and its length is not a multiple of 32 bits, padding is added to align the header on a 32-bit boundary.

**Features and Advantages of TCP:**

1. **Reliability:** TCP guarantees that data sent over the network will be delivered correctly and in the same order it was sent.
2. **Flow Control:** TCP employs flow control mechanisms to manage the rate of data transmission between devices. It prevents the sender from overwhelming the receiver with a flood of data.
3. **Congestion Control:** TCP helps prevent network congestion, which occurs when there is more data being transmitted than the network can handle.
4. **Connection-oriented:** TCP establishes a connection between the sender and receiver before data transmission. This connection-oriented nature ensures that both parties agree on the communication parameters and allows for reliable data exchange.
5. **Widely Supported:** It is the backbone of many applications and protocols that rely on reliable and ordered data delivery, such as web browsing, email, file transfer, and streaming media.

**Disadvantages of TCP:**

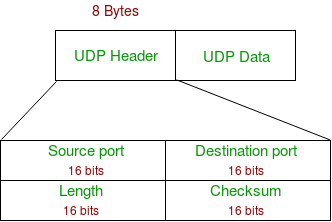
1. **Overhead:** TCP has a relatively high overhead compared to other protocols. This is due to its extensive error checking and flow control mechanisms.
2. **Latency:** TCP introduces latency in data transmission due to its acknowledgment mechanism. After sending a data segment, the sender waits for an acknowledgment from the receiver before sending the next segment.
3. **Head-of-Line Blocking:** In TCP, data segments must be received in the same order they were sent. If a single segment is lost or delayed, all subsequent segments must wait until the missing one is retransmitted.
4. **Connection-oriented:** This connection setup process involves an initial handshake, which introduces additional overhead and delay compared to connectionless protocols like UDP (User Datagram Protocol).
5. **Congestion Control:** TCP's congestion control mechanisms aim to avoid network congestion, it can lead to reduced throughput and slower data transfer rates.

**User Datagram Protocol (UDP)**

UDP (User Datagram Protocol) is a connectionless transport protocol that operates at the transport layer of the TCP/IP protocol suite. It offers a simple and lightweight alternative to TCP, providing a best-effort delivery mechanism for sending data packets between devices over IP networks.

**UDP Header:**

The UDP (User Datagram Protocol) header is a 8-byte (64-bit) structure that is added to the beginning of each UDP datagram. It contains the necessary information for the proper delivery of UDP packets. The UDP header consists of the following fields:

****

**Source Port (16 bits):** This field specifies the port number of the sending application or process.

**Destination Port (16 bits):** This field specifies the port number of the receiving application or process.

**Length (16 bits):** This field indicates the total length of the UDP datagram, including the header and the data, measured in bytes. The minimum length for a UDP packet is 8 bytes (header only).

**Checksum (16 bits):** The checksum field is used for error detection. It contains a checksum value computed over the entire UDP packet, including the header and the data. The receiving end can use this checksum to verify the integrity of the packet

**Features and Advantages of UDP:**

1. **Connectionless:** Unlike TCP, UDP is a connectionless protocol. It does not require a handshake or maintain a continuous connection between the sender and receiver. This makes UDP faster.
2. **Low Overhead:** UDP has minimal header size compared to TCP, resulting in lower overhead. This makes UDP more efficient for transmitting small amounts of data.
3. **Broadcasting and Multicasting:** UDP supports broadcasting, allowing a single packet to be sent to all devices on a network.
4. **Simple Error Handling:** UDP does not include built-in error checking or retransmission mechanisms like TCP. It allows developers to implement their own error detection and recovery mechanisms tailored to the specific requirements of the application.
5. **Control over Packet Priority:** UDP allows applications to prioritize packets based on their importance. This feature is useful in scenarios where certain data packets need to be delivered with higher priority than others.

**Disadvantages of UDP:**

1. **Lack of Reliability:** UDP does not provide built-in mechanisms for error detection, retransmission of lost packets, or congestion control. This means that packets can be lost, duplicated, or arrive out of order without any automatic recovery or retransmission.
2. **No Flow Control:** UDP does not have flow control mechanisms to manage the rate of data transmission between the sender and receiver.
3. **Inability to Guarantee Delivery:** UDP does not guarantee that packets will be delivered to the receiver. Packets can be lost or discarded in transit due to network congestion, errors, or limited buffer space
4. **Ordering of Packets:** UDP does not ensure that packets will arrive at the receiver in the same order they were sent. Packets can take different paths through the network and may arrive out of order
5. **Higher Risk of Network Congestion:** UDP does not have built-in congestion control mechanisms like TCP.

**TCP and UDP difference**

|  |  |  |
| --- | --- | --- |
|  | **TCP** | **UDP** |
| **Connection** | Connection-oriented protocol | Connectionless protocol |
| **Header Size** | TCP header size is larger (20 bytes or more) | UDP header size is smaller (8 bytes) |
| **Error Detection** | Includes extensive error detection and correction mechanisms | Limited or no error detection and correction mechanisms |
| **Reliability** | Reliable transmission with guaranteed delivery and error recovery | Unreliable transmission without guaranteed delivery or recovery |
| **Flow Control** | Supports flow control mechanisms to manage data transmission rate | Does not include built-in flow control mechanisms |

Top of Form