Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 10, October 2024

# Comparative Study of Transmission Control Protocol (TCP) and user Datagram Protocol (UDP)

Garv Rana
Computer Engineering Department
Government Polytechnic Dahod
Dahod, India

Ankita Parmar
Computer Engineering Department
Government Polytechnic Dahod Dahod,
India

Hiteshree Parikh
Computer Engineering Department
Government Polytechnic Dahod
Dahod, India

Paresh Patel
Computer Engineering Department
Government Polytechnic Dahod Dahod,
India

*Abstract* **This paper presents a comparative study of two key transport layer protocols: Transmission Control Protocol (TCP) and User Datagram Protocol (UDP). TCP is a connection-oriented protocol known for its reliability, ensuring accurate data transmission, making it ideal for applications like web browsing and file transfers. UDP, on the other hand, is connectionless and prioritizes speed over reliability, making it more suitable for real-time applications such as video streaming and gaming. The study examines the advantages, disadvantages, and real-world applications of both protocols, analyzing their performance differences and suitability in various network environments.**

*Keywords –* **Transport Layer Protocol ,User Datagram Protocol (UDP), Transmission Control Protocol (TCP ),Datagram , Segment , Applications, Three-way Handshake, comparison**

## I. INTRODUCTION

In the networking world, the two most important protocols are TCP and UDP, ruling how data would be transmitted across networks. Since the structure is connection-oriented, data will be delivered with a reliable certainty. Thus, this protocol is preferred to the applications where data accuracy is critical. On the opposite end of the spectrum, UDP has a connectionless model offering quicker communication with lesser overhead. Now a days , Quality of Service (QoS) assurance is one of the issues as the behavior of transport layer protocols affects QoS. And, most of the multimedia applications use UDP as their main transport protocol. However, for the varied QoS of distinct multimedia applications, the UDP fails to provide satisfactory results[1]. The paper is organized as follows; section 1, small explanation on transport layer protocol ; section 3, describes the transport layer protocols (TCP and UDP) in detail, while section 5, provides a comparison between the two protocols (TCP and UDP);Finally section 6, describes conclusion and future work with reference to this review work.

## II. TRANSPORT LAYER PROTOCOL

Transport layer protocol acts as an essential part of the protocol hierarchy that provides end-to-end communication between two hosts in the network[2]. Transport layer protocols provides various features such as congestion control, reliability, in sequence delivery, and flow control that contribute to performance and quality offered by the communication network. In case of multimedia applications, the correct choice of a transport layer protocol affects the quality requirements such as low delay, jitter, packet loss and throughput etc[1]. Some of the transport layer protocols are listed below [8].

- TCP(Transmission control protocol)- TCP is one of the basic protocol of the Internet protocol suite. It provides various functions such as reliable, In sequence delivery of a stream of bytes from a program on one computer to another program on the other computer
- UDP(User datagram protocol) - UDP is also one of the basic protocol of internet protocol suite. With the help of UDP a host sends the message in the form of datagram, to other hosts located on a network, without requiring to set up a transmission channel before the actual communication is started.
- DCCP(Datagram congestion control protocol) - DCCP is a transport layer protocol which is message oriented. It offers certain features like congestion.

## III. USER DATAGRAM PROTOCOL

UDP is a simple transport layer protocol that does not guarantee any reliability and in order delivery of the packets. It supports both multicasting and broadcasting [3]. A UDP packet which is called as datagram, is divided into two parts: A Header and A Data . If it detects an error in the packet, then it is declared as lost and discarded packet. This protocol is defined to make the datagram available for the packet-switched communication in an interconnected set of computer networks[4]. UDP does not provide any congestion control mechanism. The datagram format is shown below in Figure1.

The description of each field in detail is as follows [4]:

- Source port address- This field indicates the port of the sending process which sends the datagram.
- Destination port address- It indicates the port of the destination process to which the datagram is to be sent.
- Length- This field specifies the length (in bytes) of datagram which includes the header also.
- Checksum- This field is an optional 16-bit one's complement of the one's complement sum of a pseudo-IP header, UDP header, and UDP data, where the pseudo-IP header contains source IP address and destination IP address, protocol, and UDP length.

32 bit

| | |
|---|---|
| Source Port Address<br>(16 bit) | Destination Port Address<br>(16 bit) |
| Total Length<br>(16 bit) | Checksum<br>(16 it) |

Fig 1 : UDP Datagram Format [9]

## IV. BASIC OPERATION OF UDP

The operation of UDP protocol is very simple. When UDP is invoked by the application layer, the following operations are performed by UDP[5]:

- UDP encapsulates the data of users into datagrams.
- Finally forwards these datagrams to the IP layer for the transmission.

There are various applications that use UDP as their transport protocol, like Routing information protocol, Simple network management protocol, Dynamic host configuration protocol etc.[6] Voice traffic and video traffic is generally transmitted over the network by using UDP protocol. As real-time video and audio streaming protocols are designed to handle the loss of packets.

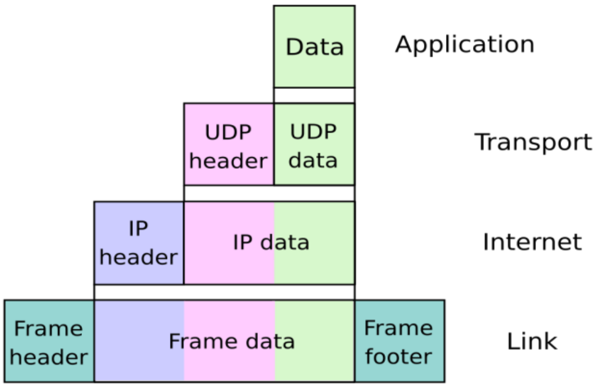


Fig – 2 UDP TRANSMISSION

## V. APPLICATION OF UDP

On the other side, these datagrams are then forwarded to UDP from the IP layer. UDP then removes the data from the datagram and forwards to the upper application layer. In UDP, a port is a number that specifies the application which is using the UDP service. It can be assumed as an address of the applications. The port number is also been used by the UDP client on the receivers end so that it can know that to which application the user data has to be forwarded[5] and below image has been given for more clarity .

## VI. COMMUNICATION OF UDP

In Fig 3 ,It Shows how UDP sends datagrams without establishing a prior connection. Unlike TCP, which uses a handshake, UDP transmits packets directly, without waiting for acknowledgments. This flow emphasizes UDP's connectionless and stateless nature, allowing for low-latency communication

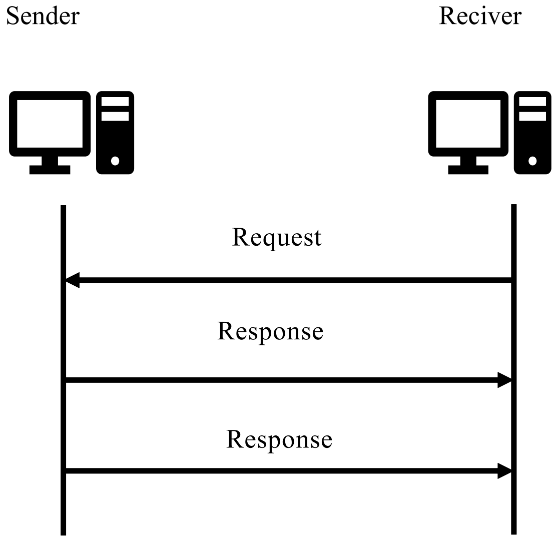
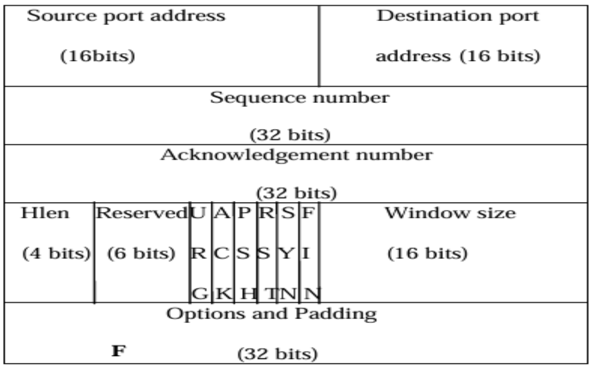


Fig 3 UDP Commnunication

## VII. TRANSMISSION CONTROL PROTOCOL

TCP is a transport layer protocol which is connection-oriented and it provides a reliable byte stream to the upper layer, called as the application layer. TCP has a mechanism based on positive acknowledgments and also provides a congestion avoidance mechanism to reduce the transmission rate when the network becomes overloaded[1]. We can also say that TCP is a time-tested transport layer protocol that provides several features like reliability, flow control and congestion control, being a heavy protocol[7]. TCP is a popular protocol which supports reliable delivery irrespective of the form of the underlying network. The datagram format is shown below in Figure4.



<table>
<tr><td colspan="9">Source port address<br>(16bits)</td><td>Destination port<br>address (16 bits)</td></tr>
<tr><td colspan="10">Sequence number<br>(32 bits)</td></tr>
<tr><td colspan="10">Acknowledgement number<br>(32 bits)</td></tr>
<tr><td>Hlen<br>(4 bits)</td><td>Reserved<br>(6 bits)</td><td>U<br>R<br>G</td><td>A<br>C<br>K</td><td>P<br>S<br>H</td><td>R<br>S<br>T</td><td>S<br>Y<br>N</td><td>F<br>I<br>N</td><td colspan="2">Window size<br>(16 bits)</td></tr>
<tr><td colspan="10">Options and Padding<br>F (32 bits)</td></tr>
</table>

Fig 4 TCP Segment Format

When comparing a TCP segment format with the UDP datagram, we can observe the key difference between the two protocols. Since UDP has smaller header frame size, UDP is faster than TCP but it lacks reliability[7]. Fig 3, shows the format of the TCP segment format. A brief description of each field is as follows:

- Source port address- The 16-bit source port number, which specifies the sender of the segment[10].

- Destination port address- The 16-bit destination port number used by sender to send the segment to the receiver[10].
- Sequence number- It is the 32 bits sequence number of the first data octet in the segment (except when SYN is present)[10].
- Acknowledgment number- It is of 32 bits, If ACK control bit is set, this field indicates the value of the next sequence number of the segment to be received. This is always required to be sent once a connection is established [10].
- Reserved- It is of 6 bits, which is reserved for future use and must be kept zero[10].
- Control bits- It is of 6 bits, there are 6 control bits listed below(from left to right)[10]:

URG: Urgent Pointer field
ACK: Acknowledgment field
PSH: Push Function
RST: Reset the connection
SYN: Synchronize sequence numbers
FIN: No more data from sender

- Window- Its size is 16 bits. It specifies the number of data octets beginning with the one indicated in the acknowledgment field which the sender of this segment is expecting to receive[10].
- Checksum- It is of 16 bits in size, This field is the 16 bit one's complement of the one's complement sum of all the 16 bit words contained in the header and text[10].
- Urgent pointer- Its size is also of 16 bits, this field communicates the current value of the urgent pointer. It indicates the sequence number of the octet following the urgent data[10].
- Options- Options are multiple of 8 bits in length and occupy the space at the end of the header. Options are included in the checksum also [10].

## VIII .APPLICATION OF TCP

TCP is used extensively in FTP for reliable file transfers, SMTP for sending emails, an HTTP/HTTPS for browsing websites. TCP ensures that all data packets arrive in sequence and without errors, making it ideal for applications that require guaranteed delivery and error checking.

## IX. BASIC OPERATION OF TCP

The operation of the TCP (Transmission Control Protocol) is more complex compared to UDP due to its connection-oriented nature. When TCP is invoked by an application layer, the following operations take place: Before any data is transmitted, TCP performs a three-way handshake to establish a reliable connection between the sender and receiver. The handshake involves three steps:

- The sender sends a SYN (synchronize) packet to the receiver.
- The receiver responds with a SYN-ACK (synchronize-acknowledgment) packet.
- The sender then sends an ACK (acknowledgment) packet, and the connection is established.

Data Transmission :-

- TCP segments the data into packets and numbers them, ensuring that they are sent and received in the correct order.

- Each packet is acknowledged by the receiver, confirming that it has been received correctly. If a packet is lost or corrupted during transmission, TCP will retransmit the lost packet.

## X. TCP THREE-WAY HANDSHAKE

In Fig 4 , It shows how TCP establishes a connection between client and server using a "SYN-SYN-ACK" process before transmitting data. This image helps clarify the reliability and connection-oriented nature of TCP

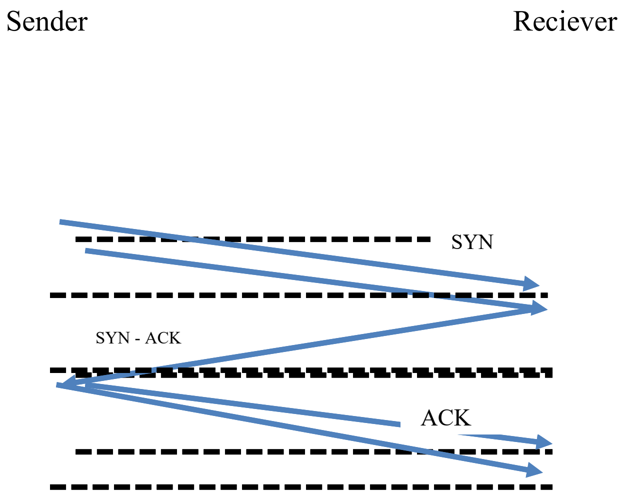


Fig 4 Three Way Handshake

## XI. Comparison Of TCP and UDP[11]

| TCP | UDP |
|---|---|
| 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. |
| 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. |
| It takes acknowledgement of data and has the ability to retransmit if the user requests. | It neither takes acknowledgement, nor it retransmits the lost data. |
| 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 |
| TCP is comparatively slower than UDP. | UDP is faster, simpler, and more efficient than TCP. |
| Flow control using sliding window protocol | There is no feature of flow control using sliding window in UDP |
| There are built-in security | UDP requires |

| | |
|---|---|
| There are built-in security features in TCP, including authentication and encryption | UDP requires application-level security because it lacks built-in security measures. |
| TCP permits the retransmission of dropped data packets. | UDP is a much faster, easier, and more effective protocol overall. |
| TCP uses congestion avoidance methods | UDP does not manage congestion. |
| A huge data stream is divided into smaller segments by the TCP segmentation procedure so that it can be sent over the network as packets. | Each message is sent as a single UDP datagram; UDP lacks any mechanisms for message segmentation or reassembling. |
| Broadcasting and Multicasting are not supported by TCP. | Broadcasting and Multicasting are supported by UDP. |
| TCP is a more complex and powerful Protocol. | UDP is straightforward and quick . |
| TCP is used by HTTP,HTTPS,FTP,SMTP, and Telnet . | UDP is used by DNS , DHCP , TFTP, SNMP,RIP, and VoIP. |

Table -1

## XII. SECURITY

1. Security Feature Of TCP

- Connection-Oriented Nature:-

TCP establishes a connection before data transmission, allowing for security measures to be implemented during the handshake[14] .

- Error Detection :-

Incorporates a checksum mechanism for error detection, ensuring data integrity. If corruption occurs, TCP can request retransmission of affected segments [14].

- Secure Sockets Layer (SSL) / Transport Layer Security (TLS):-

Operates on top of TCP to provide secure communications. Ensures data confidentialit, integrity, and authenticity through encryption. Commonly used in applications like HTTPS, FTPS, and SMTPS [14].

2. Security Challenges of UDP

- No Built-in Security:-

Lacks mechanisms for establishing connections, making it vulnerable to data interception and alteration during transmission [15] .

- Vulnerability To Attacks :-

IP Spoofing: Attackers can send packets with forged IP addresses, leading to data interception [15].

3. Securing UDP Communications :-

- Datagram Transport Layer Security (DTLS) :-

Provides security guarantees similar to TLS, but for datagram-based applications. Ensures confidentiality, integrity, and authentication while maintaining low latency characteristics.

Useful in:

- o Voice over IP (VoIP): Prevents eavesdropping and ensures data privacy.
- o Streaming Services: Protects data integrity and prevents tampering during real-time video streaming.

## CONCLUSION

In this paper, the transport layer, its various protocols and functions were reviewed. Further, the functions of TCP and UDP were described in detail and compared on the basis of their data transfer features, their basic operation and applications. Additionally, the performance of TCP in wireless networks is briefly reviewed in this paper. In this survey, it is finally concluded that, TCP protocol should be used where reliability is required such as in internet banking and UDP should be used for broadcasting and multicasting purposes like in internet gaming, internet radio etc. where some loss of packets is accepted [12]. This is why, these two protocols TCP and UDP have been considered for the survey, because TCP and UDP are the main protocols of the transport layer and each of them have a certain advantage over each other but in different cases. There are some other cases which can be considered like, UDP performs better and is suitable for applications where time is a constraint irrespective of reliable delivery, whereas TCP is suitable where time is not a constraint and reliable delivery is important. Thus, the purpose of this survey is to study and analyze the basic operation of TCP and UDP in detail, to understand the basic functions of these protocols.

## REFERENCE

[1] H. M. O. Chughtai, S. A. Malik, and M. Yousaf, “Performance Evaluation of Transport Layer Protocols for Video Traffic over WiMax,” Available: https://corenet.org.pk/data/paper48.pdf.

[2] S. Medidi, J. Ding, and M. Medidi, “Performance of Transport Protocols in Wireless Networks,” School of Electrical Engineering and Computer Science, Washington State University, Pullman, 2002.

[3] M. Rahmani, A. Pettiti, E. Biersack, E. Steinbach, and J. Hillebrand, “A Comparative Study of Network Transport Protocols for In-Vehicle Media Streaming,” BMW Group Research and Technology, Munich, Germany, 2008.

[4] J. Postel, “User Datagram Protocol,” RFC 768, USC/Information Sciences Institute, 1980.

[5] "UDP Operation”, The TCP/IP Guide. Available: http://tcpipguide.com/free/t_UDPOperation.htm.

[6] S. Zafar ,“A Survey of Transport Layer Protocols for Wireless Sensor Networks,” 2011.

[7] "User Datagram Protocol", Wikipedia. Available: http://en.wikipedia.org/wiki/User_Datagram_Protocol.

[8] "Internet Protocol Suite", Wikipedia. Available: http://en.wikipedia.org/wiki/Internet_protocol_suite.

[9] B. A. Forouzan, Data Communications and Networking, 2nd ed. New York, NY, USA: McGraw-Hill, 2003.

[10] "TCP Segment Format with Diagram", Electronics Post. Available: https://www.electronicspost.com/tcp-segment-format-with-diagram.

[11] "Transport Layer", Wikipedia. Available: http://en.wikipedia.org/wiki/Transport_layer.

[12] Chen, L., & Zhang, Y. (2023). A Comparative Analysis of TCP and UDP in Next-Generation Networks. Journal of Network and Computer Applications, 220, 102432. DOI: 10.1016/j.jnca.2023.102432.

[13] Smith, R., & Liu, J. (2022). Evaluating the Performance of Transport Layer Protocols for Video Streaming. IEEE Transactions on Multimedia, 24(4), 1001-1013. DOI: 10.1109/TMM.2022.3141517.

[14] RFC 5246: The Transport Layer Security (TLS) Protocol Version 1.2

[15] RFC 6347: Datagram Transport Layer Security (DTLS) Version 1.2