UDP Protocol

In computer networking, the UDP stands for User Datagram Protocol. The David P. Reed developed the UDP protocol in 1980. It is defined in RFC 768, and it is a part of the [TCP/IP](https://www.javatpoint.com/tcp-ip-full-form) protocol, so it is a standard protocol over the internet. The UDP protocol allows the computer applications to send the messages in the form of datagrams from one machine to another machine over the [Internet Protocol (IP)](https://www.javatpoint.com/ip) network. The UDP is an alternative communication protocol to the TCP protocol (transmission control protocol). Like TCP, UDP provides a set of rules that governs how the data should be exchanged over the internet. The UDP works by encapsulating the data into the packet and providing its own header information to the packet. Then, this UDP packet is encapsulated to the IP packet and sent off to its destination. Both the [TCP and UDP](https://www.javatpoint.com/tcp-vs-udp) protocols send the data over the internet protocol network, so it is also known as [TCP/IP](https://www.javatpoint.com/computer-network-tcp-ip-model) and UDP/IP. There are many differences between these two protocols. UDP enables the process to process communication, whereas the TCP provides host to host communication. Since UDP sends the messages in the form of datagrams, it is considered the best-effort mode of communication. [TCP](https://www.javatpoint.com/tcp) sends the individual packets, so it is a reliable transport medium. Another difference is that the TCP is a connection-oriented protocol whereas, the UDP is a connectionless protocol as it does not require any virtual circuit to transfer the data.

UDP also provides a different port number to distinguish different user requests and also provides the checksum capability to verify whether the complete data has arrived or not; the [IP](https://www.javatpoint.com/ip-full-form) layer does not provide these two services.

Features of UDP protocol

**The following are the features of the UDP protocol:**

* **Transport layer protocol**

[UDP](https://www.javatpoint.com/udp-full-form) is the simplest [transport layer communication protocol](https://www.javatpoint.com/computer-network-transport-layer). It contains a minimum amount of communication mechanisms. It is considered an unreliable protocol, and it is based on best-effort delivery services. UDP provides no acknowledgment mechanism, which means that the receiver does not send the acknowledgment for the received packet, and the sender also does not wait for the acknowledgment for the packet that it has sent.

* **Connectionless**

The UDP is a connectionless protocol as it does not create a virtual path to transfer the data. It does not use the virtual path, so packets are sent in different paths between the sender and the receiver, which leads to the loss of packets or received out of order.

**Ordered delivery of data is not guaranteed.**

In the case of UDP, the datagrams are sent in some order will be received in the same order is not guaranteed as the datagrams are not numbered.

* **Ports**

The UDP protocol uses different port numbers so that the data can be sent to the correct destination. The port numbers are defined between 0 and 1023.

* **Faster transmission**

UDP enables faster transmission as it is a connectionless protocol, i.e., no virtual path is required to transfer the data. But there is a chance that the individual packet is lost, which affects the transmission quality. On the other hand, if the packet is lost in TCP connection, that packet will be resent, so it guarantees the delivery of the data packets.

* **Acknowledgment mechanism**

The UDP does have any acknowledgment mechanism, i.e., there is no handshaking between the UDP sender and UDP receiver. If the message is sent in TCP, then the receiver acknowledges that I am ready, then the sender sends the data. In the case of TCP, the handshaking occurs between the sender and the receiver, whereas in UDP, there is no handshaking between the sender and the receiver.

* **Segments are handled independently.**

Each UDP segment is handled individually of others as each segment takes different path to reach the destination. The UDP segments can be lost or delivered out of order to reach the destination as there is no connection setup between the sender and the receiver.

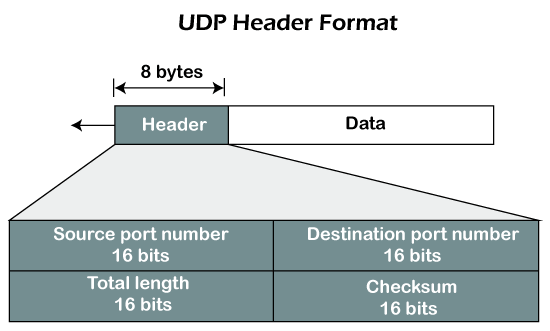
* **Stateless**

It is a stateless protocol that means that the sender does not get the acknowledgement for the packet which has been sent.

Why do we require the UDP protocol?

As we know that the UDP is an unreliable protocol, but we still require a UDP protocol in some cases. The UDP is deployed where the packets require a large amount of bandwidth along with the actual data. For example, in video streaming, acknowledging thousands of packets is troublesome and wastes a lot of bandwidth. In the case of video streaming, the loss of some packets couldn't create a problem, and it can also be ignored.

UDP Header Format

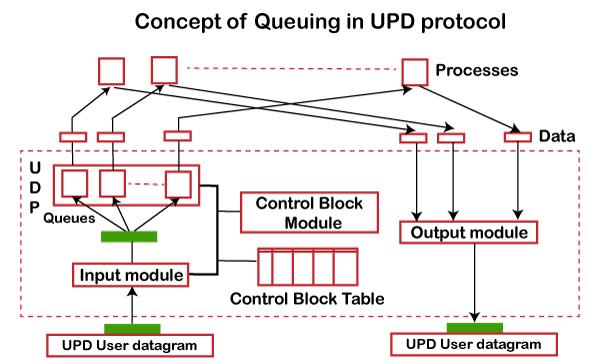


In UDP, the header size is 8 bytes, and the packet size is upto 65,535 bytes. But this packet size is not possible as the data needs to be encapsulated in the IP datagram, and an IP packet, the header size can be 20 bytes; therefore, the maximum of UDP would be 65,535 minus 20. The size of the data that the UDP packet can carry would be 65,535 minus 28 as 8 bytes for the header of the UDP packet and 20 bytes for IP header.

**The UDP header contains four fields:**

* **Source port number:** It is 16-bit information that identifies which port is going t send the packet.
* **Destination port number:** It identifies which port is going to accept the information. It is 16-bit information which is used to identify application-level service on the destination machine.
* **Length:** It is 16-bit field that specifies the entire length of the UDP packet that includes the header also. The minimum value would be 8-byte as the size of the header is 8 bytes.
* **Checksum:** It is a 16-bits field, and it is an optional field. This checksum field checks whether the information is accurate or not as there is the possibility that the information can be corrupted while transmission. It is an optional field, which means that it depends upon the application, whether it wants to write the checksum or not. If it does not want to write the checksum, then all the 16 bits are zero; otherwise, it writes the checksum. In UDP, the checksum field is applied to the entire packet, i.e., header as well as data part whereas, in IP, the checksum field is applied to only the header field.

Concept of Queuing in UDP protocol



In UDP protocol, numbers are used to distinguish the different processes on a server and client. We know that UDP provides a process to process communication. The client generates the processes that need services while the server generates the processes that provide services. The queues are available for both the processes, i.e., two queues for each process. The first queue is the incoming queue that receives the messages, and the second one is the outgoing queue that sends the messages. The queue functions when the process is running. If the process is terminated then the queue will also get destroyed.

UDP handles the sending and receiving of the UDP packets with the help of the following components:

* **Input queue:** The UDP packets uses a set of queues for each process.
* **Input module:** This module takes the user datagram from the IP, and then it finds the information from the control block table of the same port. If it finds the entry in the control block table with the same port as the user datagram, it enqueues the data.
* **Control Block Module:** It manages the control block table.
* **Control Block Table:** The control block table contains the entry of open ports.
* **Output module:** The output module creates and sends the user datagram.

Several processes want to use the services of UDP. The UDP multiplexes and demultiplexes the processes so that the multiple processes can run on a single host.

Limitations

* It provides an unreliable connection delivery service. It does not provide any services of IP except that it provides process-to-process communication.
* The UDP message can be lost, delayed, duplicated, or can be out of order.
* It does not provide a reliable transport delivery service. It does not provide any acknowledgment or flow control mechanism. However, it does provide error control to some extent.

Advantages

* It produces a minimal number of overheads.