**CSCI 367 - Computer Networks I**

UDP – User Datagram Protocol (Layer 4/Transport Layer)

References:

<https://en.wikipedia.org/wiki/User_Datagram_Protocol>

<https://www.techtarget.com/searchnetworking/definition/UDP-User-Datagram-Protocol>

The User Datagram Protocol (UDP) is a communications [protocol](https://www.techtarget.com/searchnetworking/definition/protocol) that is primarily used to establish low-latency and loss-tolerating connections between applications on the internet.

UDP speeds up transmissions by enabling the transfer of data before an agreement is provided by the receiving party. As a result, UDP is beneficial in time-sensitive communications, including voice over IP (VoIP), domain name system ([DNS](https://www.techtarget.com/searchnetworking/definition/domain-name-system)) lookup, and video or audio playback.

UDP is an alternative to Transmission Control Protocol ([TCP](https://www.techtarget.com/searchnetworking/definition/TCP)). Both UDP and TCP run on top of IP and are sometimes referred to as UDP/IP or [TCP/IP](https://www.techtarget.com/searchnetworking/definition/TCP-IP). However, there are important differences between the two. For example, UDP enables process-to-process communication, while TCP supports host-to-host communication.

TCP sends individual [packets](https://www.techtarget.com/searchnetworking/definition/packet) and is considered a reliable transport medium. On the other hand, UDP sends messages, called *datagrams*, and is considered a best-effort mode of communications. This means UDP doesn't provide any guarantees that the data will be delivered or offer special features to retransmit lost or corrupted messages.

UDP provides two services not provided by the IP layer. It provides [port numbers](https://www.techtarget.com/searchnetworking/definition/port-number) to help distinguish different user requests. It also provides an optional [checksum](https://www.techtarget.com/searchsecurity/definition/checksum) capability to verify that the data arrived intact.

**User Datagram Protocol features**

User Datagram Protocol has attributes that make it beneficial for use with applications that can tolerate lost data. Below are some examples:

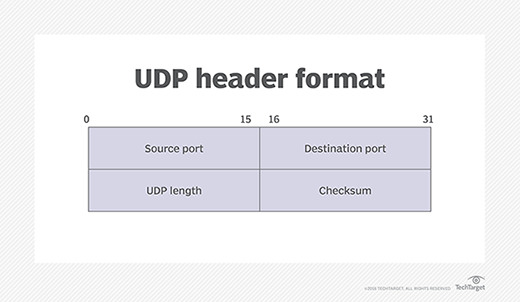
* It allows packets to be dropped and received in a different order than they were transmitted, making it suitable for real-time applications where latency might be a concern.
* It can be used for transaction-based protocols, such as DNS or Network Time Protocol ([NTP](https://www.techtarget.com/searchnetworking/definition/Network-Time-Protocol)).
* It can be used where a large number of clients are connected and where real-time error correction isn't necessary, such as gaming, voice or video conferencing, and streaming media.

**UDP header composition**

UDP uses headers when packaging message data to transfer over network connections. UDP headers contain a set of parameters -- called *fields* -- defined by the technical specifications of the protocol.

The User Datagram Protocol header has four fields, each of which is 2 bytes. They are the following:

* **Source Port**: The application port number of the sender.
* **Destination Port**: The application port number of the receiver to which the datagram is addressed.
* **Length**: The length in bytes of the UDP header and any encapsulated data.
* **Checksum**: Used in error checking -- its use is required in [IPv6](https://www.techtarget.com/searchnetworking/definition/IPv6-Internet-Protocol-Version-6) and optional in [IPv4](https://www.techtarget.com/whatis/definition/IPv4-address-class).



**How UDP works**

UDP uses IP to get a datagram from one computer to another. UDP works by gathering data in a UDP packet and adding its own header information to the packet. This data consists of the source and destination ports on which to communicate, the packet length and a checksum. After UDP packets are encapsulated in an IP packet, they're sent off to their destinations.

Unlike TCP, UDP doesn't guarantee the packets will get to the right destinations. This means UDP doesn't connect to the receiving computer directly, which TCP does. Rather, it sends the data out and relies on the devices in between the sending and receiving computers to correctly get the data where it's supposed to go.

Most applications wait for any replies they expect to receive as a result of packets sent using UDP. If an application doesn't receive a reply within a certain time frame, the application sends the packet again, or it stops trying.

UDP uses a simple transmission model that doesn't include [handshaking dialogues](https://www.techtarget.com/searchnetworking/answer/Which-is-most-secure-CHAP-or-PAP) to provide reliability, ordering or data integrity. Consequently, UDP's service is unreliable. Packets may arrive out of order, appear to have duplicates or disappear without warning.

Although this transmission method doesn't guarantee that the data being sent will reach its destination, it does have low overhead and is popular for services that don't absolutely have to work the first time.