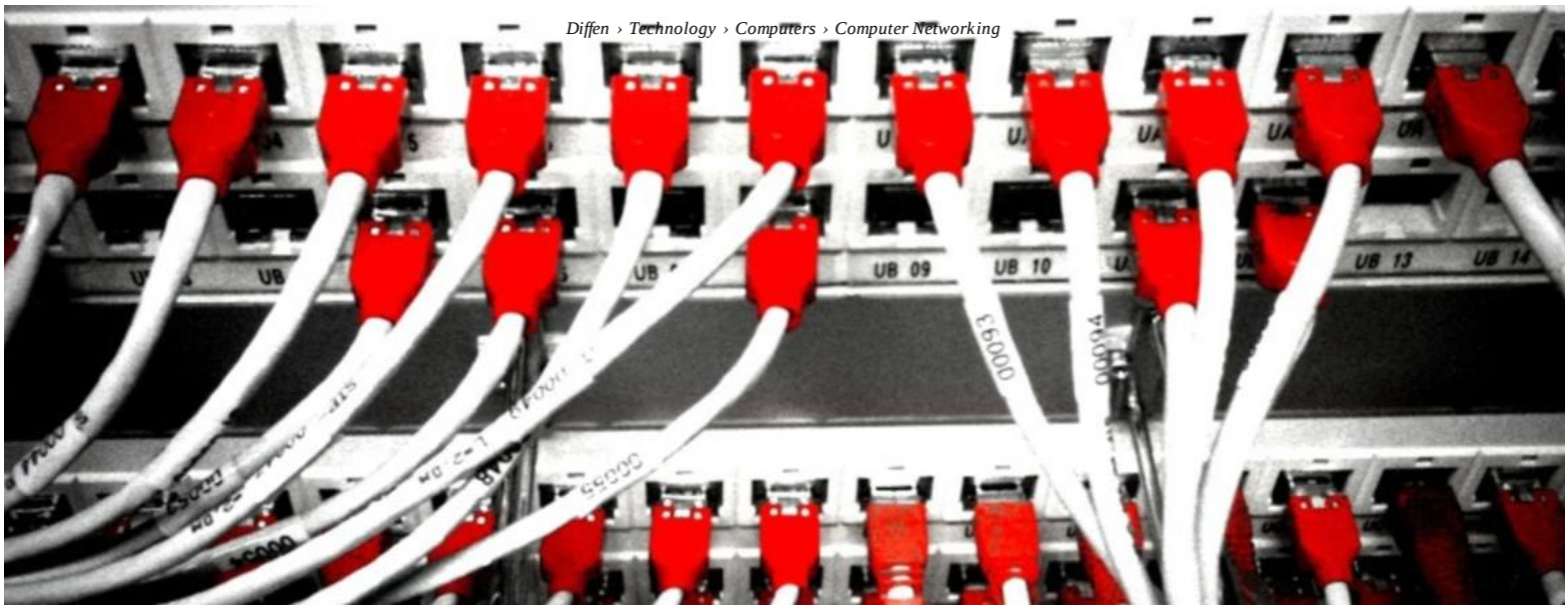
TCP *vs* UDP



There are two types of Internet Protocol (IP) traffic. They are **TCP** or **Transmission Control Protocol** and **UDP** or **User Datagram Protocol**. TCP is connection oriented – once a connection is established, data can be sent bidirectional. UDP is a simpler, connectionless Internet protocol. Multiple messages are sent as packets in chunks using UDP.

Comparison chart

Differences — **Similarities** —

	TCP	UDP
Acronym for	Transmission Control Protocol	User Datagram Protocol or Universal Datagram Protocol
Connection	TCP is a connection-oriented protocol.	UDP is a connectionless protocol.
Function	As a message makes its way across the internet from one computer to another. This is connection based.	UDP is also a protocol used in message transport or transfer. This is not connection based which means that one program can send a load of packets to another and that would be the end of the relationship.
Usage	TCP is suited for applications that require high reliability, and transmission time is relatively less critical.	UDP is suitable for applications that need fast, efficient transmission, such as games. UDP's stateless nature is also useful for servers that answer small queries from huge numbers of clients.
Examples	HTTP, HTTPs, FTP, SMTP, Telnet	DNS, DHCP, TFTP, SNMP, RIP, VOIP.
Ordering of data packets	TCP rearranges data packets in the order specified.	UDP has no inherent order as all packets are independent of each other. If ordering is required, it has to be managed by the application layer.
Speed of transfer	The speed for TCP is slower than UDP.	UDP is faster because there is no error-checking for packets.
Reliability	There is absolute guarantee that the data transferred remains intact and arrives in the same order in which it was sent.	There is no guarantee that the messages or packets sent would reach at all.

Header Size	TCP header size is 20 bytes	UDP Header size is 8 bytes.
Common Header Fields	Source port, Destination port, Check Sum	Source port, Destination port, Check Sum
Streaming of data	Data is read as a byte stream, no distinguishing indications are transmitted to signal message (segment) boundaries.	Packets are sent individually and are checked for integrity only if they arrive. Packets have definite boundaries which are honored upon receipt, meaning a read operation at the receiver socket will yield an entire message as it was originally sent.
Weight	TCP is heavy-weight. TCP requires three packets to set up a socket connection, before any user data can be sent. TCP handles reliability and congestion control.	UDP is lightweight. There is no ordering of messages, no tracking connections, etc. It is a small transport layer designed on top of IP.
Data Flow Control	TCP does Flow Control. TCP requires three packets to set up a socket connection, before any user data can be sent. TCP handles reliability and congestion control.	UDP does not have an option for flow control
Error Checking	TCP does error checking	UDP does error checking, but no recovery options.
Fields	1. Sequence Number, 2. AcK number, 3. Data offset, 4. Reserved, 5. Control bit, 6. Window, 7. Urgent Pointer 8. Options, 9. Padding, 10. Check Sum, 11. Source port, 12. Destination port	1. Length, 2. Source port, 3. Destination port, 4. Check Sum
Acknowledgement	Acknowledgement segments	No Acknowledgment
Handshake	SYN, SYN-ACK, ACK	No handshake (connectionless protocol)

Contents

Differences in Data Transfer Features

TCP ensures a reliable and ordered delivery of a stream of bytes from user to server or vice versa. **UDP** is not dedicated to end to end connections and communication does not check readiness of receiver.

.

Reliability

TCP is more reliable since it manages message acknowledgment and retransmissions in case of lost parts. Thus there is absolutely no missing data. **UDP** does not ensure that communication has reached receiver since concepts of acknowledgment, time out and retransmission are not present.

Ordering

TCP transmissions are sent in a sequence and they are received in the same sequence. In the event of data segments arriving in wrong order, TCP reorders and delivers application. In the case of **UDP**, sent message sequence may not be maintained when it reaches receiving application. There is absolutely no way of predicting the order in which message will be received.

Connection

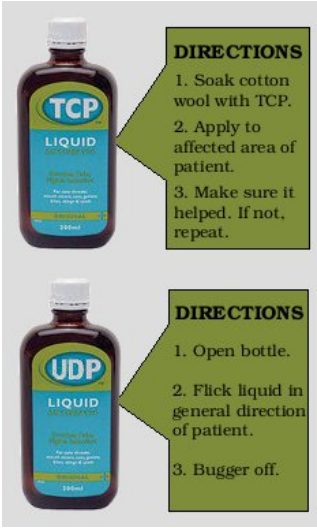
TCP is a heavy weight connection requiring three packets for a socket connection and handles congestion control and reliability. **UDP** is a lightweight transport layer designed atop an IP. There are no tracking connections or ordering of messages.

Method of transfer

TCP reads data as a byte stream and message is transmitted to segment boundaries. **UDP** messages are packets which are sent individually and on arrival are checked for their integrity. Packets have defined boundaries while data stream has none.

How TCP and UDP work

A TCP connection is established via a three way handshake, which is a process of initiating and acknowledging a connection. Once the connection is established data transfer can begin. After transmission, the connection is terminated by closing of all established virtual circuits.



A light-hearted look at the philosophical difference between UDP and TCP

UDP uses a simple transmission model without implicit hand-shaking dialogues for guaranteeing reliability, ordering, or data integrity. Thus, UDP provides an unreliable service and datagrams may arrive out of order, appear duplicated, or go missing without notice. UDP assumes that error checking and correction is either not necessary or performed in the application, avoiding the overhead of such processing at the network interface level. Unlike TCP, UDP is compatible with packet broadcasts (sending to all on local network) and multicasting (send to all subscribers).

Different Applications of TCP and UDP

Web browsing, email and file transfer are common applications that make use of TCP. TCP is used to control segment size, rate of data exchange, flow control and network congestion. TCP is preferred where error correction facilities are required at network interface level. UDP is largely used by time sensitive applications as well as by servers that answer small queries from huge number of clients. UDP is compatible with packet broadcast - sending to all on a network and multicasting – sending to all subscribers. UDP is commonly used in Domain Name System, Voice over IP, Trivial File Transfer Protocol and online games.

Related Comparisons

Hub *vs* Switch

Internet *vs* World Wide Web

Router *vs* Switch

Modem *vs* Router

References

- http://en.wikipedia.org/wiki/Transmission_Control_Protocol
- http://en.wikipedia.org/wiki/User_Datagram_Protocol

Comments: TCP vs UDP

86 comments

Add a comment...

Comment using...

Fresnel Erlang
thanks... very useful...
Reply · 1 · Like · October 31 at 10:48pm

Usman Ali Butt
Very helpful thanks.
Reply · 3 · Like · August 27 at 3:39am

JP Shepherd · Student at ChristLife Ministry
cool! I get the meaning now . thanks
Reply · 1 · Like · July 10 at 5:34pm

aaaa · Karachi, Pakistan
very useful) :
Reply · 1 · Like · June 28 at 11:43pm

Jeremiah Seni · Mzumbe University
thank you
Reply · 2 · Like · April 21 at 1:44pm

Krishna Singh · Asst. Professor at Rgukt
Good works, thanx.
Reply · 1 · Like · May 12 at 1:00am

Jenny Parton · Account Executive at Classic Rock 95.1
Thank you!
Reply · 1 · Like · March 7 at 2:13pm

Timothy Dominique · Nursing at Choice Rehab and care center
That was helpful, thanks. yes ports, ports and more ports.
Reply · Like · February 15 at 1:43pm

Caliph Nourredine · Kampala, Uganda
It was Helpful. Thanks.
Reply · Like · February 17 at 3:31am

Sivanantham Mca · College of Engineering, Guindy, Anna University
it is good to learn deference between tcp and udp.....
Reply · Like · February 11 at 7:17am

Clubmaster Steele
I'm bookmarking this. Very helpful. Thank you,
Reply · Like · February 6 at 3:51pm

NiNa Hassan · Supervisor at New Sulaimany University Campus
really helpful I was looking for this one.
Reply · 1 · Like · December 10, 2012 at 2:05pm

Shahbaz Dogar · Brunel University
well done nina
Reply · 1 · Like · December 10, 2012 at 5:22pm

Ashwini Selukar
perfect
Reply · Like · January 20 at 4:39am

Oligarch' RussMatazz Maruping · Gaborone, Botswana
very helpful and well explained, great work

very helpful and well explained, great work.
Reply · Like · January 6 at 12:27pm

Rhoda Coburn · John Marshall High School
excellent explanation and presedntation of information.
Reply · 1 · Like · November 15, 2012 at 7:04am

Achithra Prasad Sachinthaka · Open University
great explanation..
Reply · Like · November 29, 2012 at 4:00am

Benjamin Brown · University for Development Studies
this is very useful information and I am very greatful.
Reply · Like · November 9, 2012 at 9:25am

Messiah-akwesi Afful · University for Development Studies
oh Tumu
Reply · Like · November 9, 2012 at 11:21pm

Rashedul Islam Rasel · CEO & Founder at RSAsoft BD
thank you very much...
Reply · Like · November 9, 2012 at 8:44pm

Josie Kabia · Maseno University
BOOOOOOOORING.....WOT ON EARTH WAS MY LECTURER THINKING!
Reply · Like · November 1, 2012 at 12:02pm

Varsha Singh · M.g.c.p.s.
thanx for providede valuable notes.
Reply · Like · September 22, 2012 at 11:35am

Ankita Chawla
niceeeeeeeeeeeeee
Reply · 3 · Like · May 18, 2012 at 12:17am

Nauman Zafar · KICSIT
gr8
Reply · 2 · Like · April 28, 2012 at 11:44am

Husam Nogaim · Works at الأمين العام للمفسبكين اليمنيين
it's so nice.
Reply · 1 · Like · April 15 at 1:08am

Nắng Mùa Đông · Top Commenter · Hanoi, Vietnam
thanks so much. i nêd it
Reply · Like · November 29 at 11:14am

View 57 more

Facebook social plugin

Diffen

About Diffen
Sign up
User Requests
Feedback
Comparison Categories

Stay connected

Comparison Categories
Actors
Actresses
Bands
Baseball Players
Billionaires
Cameras
Cars
Cell Phones
Cities
Companies
Countries
Credit Cards
Cricketers
Fictional Characters
Formula 1 Drivers
GPS Systems
Music Genres
Musicians
Processors
Soccer Players
Tennis Players