Sveučilište Josipa Jurja Strossmayera u Osijeku

Odjel za matematiku

Preddiplomski studij matematike

*Seminarski rad iz predmeta*

**Računalne mreže i usluge**

*Naslov rada:*

**TCP i UDP protokoli**

Čordaš Rebeka, 802

Čorić Zoran, 824

Đumić Mateja, 792

Strišković Jelena, 803

Osijek, svibanj 2011.

Sadržaj

[1. Uvod - 3 -](#_Toc292665932)

[2. UDP protokoli - 3 -](#_Toc292665933)

[2.1. Općenito o UDP-u - 3 -](#_Toc292665934)

[2.2. Kako UDP funkcionira? - 3 -](#_Toc292665935)

[2.3. Zašto koristiti UDP? - 3 -](#_Toc292665936)

[2.4. Struktura paketa u UDP-u - 5 -](#_Toc292665937)

[3. TCP protokoli - 6 -](#_Toc292665938)

[3.1. TCP općenito - 6 -](#_Toc292665939)

[3.2. Uspostava veze - 7 -](#_Toc292665940)

[3.3. Razmjena podataka - 7 -](#_Toc292665941)

[3.4. TCP portovi - 9 -](#_Toc292665942)

[4. TCP i UDP - usporedba - 9 -](#_Toc292665943)

[5. Zaključak - 11 -](#_Toc292665944)

[Literatura - 12 -](#_Toc292665945)

# Uvod

Kada govorimo o računalnim mrežama neizbježno je spomenuti i protokole koji se prilikom „izgradnje“ mreže koriste, jer upravo pomoću njih čvorovi računalne mreže ostvaruju komunikaciju te prijenos podataka između sebe. Jedni od važnijih protokola u ostvarenju prijenosa su zasigurno UDP i TCP protokol. U ovom seminarskom radu pokušat ćemo vas upoznati sa tim protokolima kroz pregled njihovih osnovnih karakteristika i načina funkcioniranja. Nakon samog upoznavanja sa navedenim protokolima, pokušat ćemo ih usporediti i doći do zaključka zašto je potrebno postojanje oba, a ne samo jedan od njih, te u kojoj situaciji koristiti TCP, a u kojoj UDP protokol.

# UDP protokoli

## Općenito o UDP-u

UDP (User Datagram Protocol) je jedan od glavnih članova paketa internetskih protokola (Internet Protocol Suite), tj. paketa protokola koji se koriste na internetu. Posebnost UDP-a je u tome što uz njegovu pomoć računalne aplikacije mogu slati poruke (datagrame) drugim korisnicima, tj. aplikacijama na internetu bez posebnog postavljanja kanala i unaprijed postavljenih puteva. UDP protokol je osmislio David P. Reed 1980. godine i formalno definirao kao RFC 768 protokol.

## Kako UDP funkcionira?

Gledajući ostale internetske protokole UDP radi samo osnovni minimum koji neki transportni protokol treba. Ako zanemarimo multipleksiranje i demultipleksiranje funkcija te osnovni oblik provjere grešaka, UDP ne dodaje ništa IP-u. Što znači da programer koji se odluči da će njegova aplikacija komunicirati pomoću UDP protokola umjesto TCP-a pravi aplikaciju koja „priča“ direktno s udaljenim IP-jem. UDP uzima poruku od te aplikacije, za nju „prikači“ izvorni i krajnji broj polja porta (za multipleksiranje i demultipleksiranje), doda dva manje važna polja i to sve prenese na mrežu. Dalje mreža od te poruke napravi IP datagram i najbolje što može pokuša tu poruku dostaviti udaljenom poslužitelju (korisniku). Ukoliko poruka stigne do poslužitelja, UDP koristi broj porta IP poslužitelja i njegovu adresu za dostavu podataka, u malim paketima, željenoj aplikaciji.

## Zašto koristiti UDP?

UDP se preferira u nekim aplikacijama zbog sljedećih razloga:

* **Nema čekanja na uspostavu konekcije.** Što znači da UDP prije nego počne slati podatke samo pošalje te podatke bez ikakvih provjera i predprovjera. Kod njega nema čekanja na uspostavu veze, što je vjerojatno jedan od glavnih razloga zašto DNS koristi UDP umjesto TCP-a, dok HTTP koristi TCP protokol zato što je pozdanost ključna za web stranice s tekstom. (više o samom TCP-u u idućem poglavlju)
* **Nema „stanja“ veze.** To bi značilo da za razliku od TCP-a, koji održava vezu sa klijentima da bi implementirao svoj sigurni prijenos podataka, UDP ne održava vezu i ne prati nijedan parametar koji prati TCP. Zbog toga UDP može održavati više aktivnih klijenata od TCP-a.
* **Malo zaglavlje.** TCP ima 20 bajtova zaglavlja u svakom dijelu, dok UDP ima samo 8 bajtova.
* **Neregulirana stopa slanja.** TCP ima kontrolni mehanizam koji „guši“ vezu kada ima više od jedne veze između primatelja i pošiljatelja. To zagušenje veze ima velik utjecaj na aplikacije koje stvari obavljaju u realnom vremenu, zato što one mogu tolerirati određen gubitak podataka ali zahtjevaju određenu brzinu primanja istih. Za razliku od TCP-a UDP šalje podatke onoliko brzo koliko ih aplikacija stvara, jedini utjecaj na brzinu slanja ima brzina početne internet veze i hardverske mogućnosti poslužitelja i korisnika (procesor, rami...).

Ovo sve znači da je ugrađena nezahtjevnost UDP-a vrlo korisna kod servera koji poslužuju puno klijenata. Za razliku od TCP-a, UDP je kompatibilan sa slanjem podataka svima na lokalnoj mreži i multikastingu (slanje svima određenima).



**Slika 1.** Pregled najčešće korištenih portova

(preuzeto s: <http://www.informatika.buzdo.com/s916.htm>)

## Struktura paketa u UDP-u



**Slika 2.** Struktura paketa u UDP-u (preuzeto s: <http://en.wikipedia.org/wiki/User_Datagram_Protocol>)

Zaglavlje UDP-a se sastoji od 4 polja, svako od tih polja ima 2 bajta (16 bitova). Njihovo korištenje je neobavezno u Ipv4 (na slici 2. označeno rozom bojim). U Ipv6 samo je odredišni port neobavezan.

**Source port number -** ovo polje identificira broj porta pošiljatelja kada je potrebno i uzima se da je to port na koji se odgovara, ukoliko je to potrebno, a ako nije potrebno onda se uzima da je 0. Ako se tu radi o klijentu onda će broj porta biti neki otvoreni port, a ako je to poslužitelj onda će broj porta vjerojatno biti jedan od unaprijed određenih brojeva. (vidi sl.1.)

**Destination port number -** identificira pošiljateljev port i obavezan je. Slično kao i u prošloj stavci ako se tu radi o klijentu onda će broj porta biti neki otvoreni port, a ako je to poslužitelj onda će broj porta vjerojatno biti jedan od unaprijed određenih brojeva. (vidi sl.1.)

**Length -** polje koje određuje dužinu cijelog datagrama u bajtovima. Minimalna dužina je 8 bajta zato što je to dužina zaglavlja. Teoretski limit je 65 535 bajta (8 bajta zaglavlje + 65 527 bajta za podatke) za UDP datagram. Praktični limit za veličinu je ograničen sa Ipv4 protokolom na 65507 bajta (65 535 − 8 bajta UDP zaglavlje − 20 bajta IP zaglavlje).

**Checksum - o**vo polje se koristi za provjeru grešaka zaglavlja i podataka. Ukoliko pošiljatelj ne pošalje tu vrijednost, za vrijednost polja se uzimaju sve nule.

# TCP protokoli

## TCP općenito

TCP (Transmission Control Protocol) jedan je od osnovnih protokola unutar IP grupe protokola kojeg je postavio DoD (Department of Defense) USA 1977. On je dominantan, spojni (korištenjem TCP protokola, aplikacija na nekom od hostova umreženog u računalnu mrežu kreira virtualnu konekciju prema drugom hostu te putem te ostvarene konekcije zatim prenosi podatke), prijenosni protokol interneta.

Kao što i samo ime kaže on kontrolira i osigurava pouzdanu isporuku podataka do odredišta u kontroliranom redoslijedu (kada neki dio poruke ne stigne u pravilnom redoslijedu, server šalje klijentu poruku u kojoj zahtjeva da se taj dio poruke ponovo pošalje i tada se svi neporedani dijelovi moraju ponovo poslagati – treba više vremena, ali smo sigurni da su podaci stigli). Osim toga, TCP pruža i mogućnost višestrukih istovremenih konekcija prema jednoj aplikaciji na jednom hostu od strane više klijenata, gdje su najčešći primjeri za to web ili poslužitelji e-pošte.

Dakle, osnovna svojstva usluge koju nudi TCP su:

* Pouzdanost
* Veza od točke to točke
* Dvosmjerni prijenos podataka
* Svi podaci tretiraju se kao niz okteta

|  |
| --- |
| APLIKACIJSKI SLOJ |
| PRIJENOSNI SLOJ |
| INTERNET SLOJ |
| SLOJ PRISTUPA MREŽI |

**Slika 3.** Arhitektura TCP/IP protokola.- TCP je protokol prijenosnog sloja TCP/IP modela, a omogućava pouzdan prijenos podataka između dva krajnja uređaja

## Uspostava veze

Pri uspostavi veze proces koji se izvodi na jednom računalu (klijent) želi uspostaviti vezu s procesom na nekom drugom računalu (server, poslužitelj).



**Slika 4.** "Three-way handshake" (preuzeto s: <http://condor.depaul.edu/jkristof/technotes/tcp-connection-establishment.jpg>)

Klijentski proces informira klijentski TCP da želi uspostaviti vezu s poslužiteljem. Klijentsko računalo tada šalje poslužitelju prvi specijalni segment tzv. SYN bit. Poslužitelj odgovara drugim specijalnim TCP segmentom i konačno klijent odgovara trećim specijalnim segmentom. Ova procedura se naziva "three-way handshake". Kada se sva tri koraka obave, klijent i poslužitelj jedan drugome mogu slati segmente koji sadrže podatke. U svakom budućem segmentu SYN će biti postavljen na 0.

## Razmjena podataka

TCP entiteti razmjenjuju podatke u obliku segmenata. Segment se sastoji od zaglavlja koje ima 20 okteta za kojim slijedi nula ili više okteta podataka. Segmenti koji imaju 0 byta podataka koriste se za potvrde i kontrole poruke. Veličina segmenta je varijabilna uz dva ograničenja:

* Svaki segment uključujući i TCP zaglavlje mora stati u 65 535 okteta IP paketa
* Svaka mreža ima svoj MTU (Maximum Transmission Unit), a to je najveća dopuštena jedinica za prijenos koja definira gornju granicu veličine segmenta.

Ako je segment prevelik za mrežu kroz koju mora proći, čvor vrši fragmentaciju u više manjih segmenata od kojih svaki dobiva svoje IP zaglavlje. Osnovni protokol kojeg koriste TCP entiteti je protokol s klizajućim prozorom (engl. Sliding Window): Nakon slanja segmenta predajnik pokreće brojač (engl. timer). Kad segment stigne na odredište, prijemnik šalje u segmentu potvrdu s brojem jednakim slijedećem broju segmenta kojeg očekuje. Ako brojač istekne prije nego što je primljena potvrda, segment se šalje ponovno (ako server ne dobije dio poruke on će zatražiti od klijenta da mu ponovo pošalje dio koji nije stigao i tako je osiguran integritet poruke. Također kao što je već rečeno kada neki dio poruke ne stigne u pravilnom redoslijedu, server šalje klijentu poruku u kojoj zahtjeva da se taj dio poruke ponovo pošalje i tada se svi neporedani dijelovi moraju ponovo poslagati).



**Slika 5.** Struktura TCP segmenta

(Ideja slike preuzeta s: <http://azurit.blog.matfyz.sk/tcp_segment.jpg>)

U zaglavlju prvo idu polja za broj usluge na predajnoj (source port) i prijemnoj strani (destination port). Nakon toga slijedi polje „slijedni broj“ (sequence number) u kojem je zapisan redni broj prvog byta u tom segmentu jer svaki byte kod TCP veze ima svoj broj. Polje „broj potvrde“ (acknowledgement number) određuje slijedeći byte koji se očekuje (važno za pravilan dolazak paketa). Polje „dužina zaglavlja“ (TCP header length) kaže koliko 32- bitnih riječi ima zaglavlje. 6-bitno polje se ne koristi (na slici praznina između dužina zaglavlja i zastavice), a nakon njega slijedi 6 jednobitnih zastavica: URG-za hitno slanje, ACK-potvrda primitka, PSH-pražnjenje spremnika (buffer), RST-resetiranje veze, SYN-sinkronizacija i FIN-oslobađanje veze. Polje “veličina prozora” (window size) kaže koliko se bytova može poslati počevši od potvrdnog byta. “Polje za provjeru“ (checksum) provjerava ispravnost zaglavlja i podataka koji slijede. Polje “pokazivača hitnosti“ (urgent pointer) pokazuje na važne podatke (zastavica URG je tada postavljena u 1). Polje „opcije“ ( options) služi za uključivanje dodatnih mogućnosti kojih inače nema u zaglavlje npr. računalo može navesti maksimalnu dozvoljenu veličinu TCP segmenta koje prihvaća.

## TCP portovi

***Opće poznati portovi*** (engleski *well known ports*) su dodijeljeni od strane Internet Assigned Numbers Authority, organizacije koja se brine za IP adresni prostor, vršne domene te druge detalje vezane uz IP protokol. Neke od najpoznatijih aplikacija koje svakodnevno susrećemo koriste TCP. To su primjerice: FTP (TCP port 21), Telnet (port 23), SMTP (port 25) i HTTP (port 80). Primjećujemo da se TCP protokol koristi kod aplikacija u kojima je bitna povratna informacija. Npr. kod e-maila (SMTP) je bitno da server dobije povratnu informaciju je li klijent dobio sve informacije koje je zatražio. Također vidimo da se koristi pri uspostavi dvosmjernog 8-bitnog komunikacijskog kanala između dva umrežena računala (Telnet). TCP protokol rabi portove od 1 do 65535, tj. ukupno ima 65535 mogućih različitih portova.

# TCP i UDP - usporedba

U prethodna dva dijela smo saznali osnovne karakteristike TCP i UDP protokola. Svaki od tih protokola ima svoje prednosti i mane. Većina ljudi će pomisliti da UDP protokol i nije potreban kada se TCP protokol brine o svemu i kada je puno sigurnije da će poslani podaci stići na odredište ako o tome brine TCP protokol. No i UDP ima svojih prednosti. On se koristi u prijenosima podataka u kojima nije toliko bitno da su baš svi paketi stigli na odredište. Npr. kod live streaming televizijskih programa, VoIP – a (Voice over Internet Protocol), DNS – a (Domain Name Server) itd. Sve prednosti i mane TCP –a i UDP – a kao i njihovu usporedbu najlakše ćemo vidjeti ako ih posložimo u tablicu.



**Slika 6.** Usporedba karakteristika TCP i UDP protokola

(slika preuzeta s:[*http://www.theitstuff.com/files/2009/01/7\_tcp\_vs\_udp-300x233.jpg*](http://www.theitstuff.com/files/2009/01/7_tcp_vs_udp-300x233.jpg)*)*

Kao što vidimo iz tablice TCP je pouzdan protokol, a UDP nepouzdan. Razlog tome je što kada pošaljemo poruku putem TCP socketa sigurni smo da će ona stići na odredište, osim ako dođe do potpunog prekida veze; ako server ne dobije dio poruke on će zatražiti od klijenta da mu ponovo pošalje dio koji nije stigao i tako je osiguran integritet poruke. UDP ne brine o takvim stvarima. On jednostavno šalje poruke i nakon što ih je poslao ne brine što se s njima događa.

Nadalje, vidimo da se TCP zasniva na pravoj vezi između klijenta i servera, a UDP nema vezu. Kada se koristi UDP, klijent može primati poruke od bilo koga, dok kod TCP –a klijent može primati poruke samo od servera s kojim je ostvarena konekcija (odnosno veza).

TCP protokol brine o redoslijedu kojim podaci dolaze na odredište. Ako pošaljemo neka dva podatka jedan za drugim TCP nam osigurava da će oni tim istim redoslijedom stići na odredište jer svaki podatak u zaglavlju ima pohranjen ID i redni broj i kada podaci stižu na odredište server provjeri kojim bi redoslijedom trebali biti poslagani i tako ih procesuira. Kod UDP–a nikad ne možemo biti sigurni hoće li podaci u pravilnom redoslijedu stići na odredište, no i to mi kao programeri možemo zaobići jednostavno numerirajući podatke i provjeravajući na serveru jesu li podaci došli u pravilnom redoslijedu.

Sada kad smo vidjeli usporedbu prednosti i mana TCP i UDP protokola možemo vidjeti i gdje se koriste TCP, a gdje UDP protokoli.



**Slika 7.** Aplikacije u kojima se koriste TCP i UDP protokoli

(slika preuzeta s: *http://www3.gdin.edu.cn/jpkc/dzxnw/jsjkj/chapter3/33.htm)*

Iz priložene tablice vidimo da se TCP protokol koristi kod aplikacija u kojima je bitna povratna informacija. Npr. kod e-maila je bitno da server dobije povratnu informaciju je li klijent uspio dobiti sve informacije koje je zatražio, pa ako klijent nije uspio dobiti sve informacije o e-mailu koji je htio pročitati server automatski šalje podatke koji nisu stigli klijentu. Isto tako je bitno kada otvaramo neku stranicu u našem web browseru da nam se otvori cijela stranica, a ne samo neki njeni dijelovi i upravo se o tome brine TCP. No, vidimo i da UDP ima jako široku primjenu. Npr. kada bi u internet telefoniji za razmjenu podataka bio odgovoran TCP protokol i kada bi došlo do gubitka nekog paketa server bi tražio od klijenta da ponovo pošalje taj paket što bi dovelo do velikih nesrazmjera u vremenu slanja i primanja poruke. Zato je dobro da se o tome brine UDP jer on ne radi takve provjere, a u ovom slučaju ako se i izgubi koji paket putem neće doći do većih problema jer su paketi veoma mali pa će gubitci biti gotovo neprimjetni.

# Zaključak

TCP i UDP su protokoli koji omogućuju prijenos podataka unutar mreže. Njihova najveća razlika, a ujedno i razlog potrebe postojanja oba od njih, je potvrda, odnosno izostanak potvrde, primitka poruke, te „održavanje“ veze između klijenta i poslužitelja.

Različiti zadaci, iziskuju i različit protokole, tako da prilikom programiranja, programer mora znati koji od ovih dva je potrebno koristiti. Odluka je ponekad vrlo laka, a ponekad je potrebno razmotriti prednosti i nedostatke ovih protokola, prvenstveno koliko će naša aplikacija biti „ugrožena“ ako dođe do gubitka nekog dijela poruke, a s druge strane koliko će poslužitelj zbog uspostave same veze putem TCP protokola biti „zatrpan“.

Kako bi mogli donijeti odluku o korištenju određenog protokola, ali i razumjeti stvari koje se događaju prilikom korištenja tih protokola, potrebno je razumjeti način funkcioniranja samih protokola, a ovim seminarskim radom mislimo da smo uspjeli iznijeti ono najosnovnije ali i najvažnije o TCP i UDP protokolima.

# Literatura

1. Kurose, Ross: Computer Networking - A Top-down Approach Featuring the Internet, 3rd Ed, Addison-Wesley
2. **http://en.wikipedia.org/wiki/Transmission\_Control\_Protocol** - 25.4.2011. - 15:40
3. **http://www.lincoln.edu/math/rmyrick/ComputerNetworks/InetReference/83.htm** - 25.4.2011. - 15:41
4. **http://www.lincoln.edu/math/rmyrick/ComputerNetworks/InetReference/85.htm** - 25.4.2011. - 15:42
5. **http://en.kioskea.net/contents/internet/tcp.php3** - 25.4.2011. - 15:44
6. **http://searchnetworking.techtarget.com/definition/TCP** - 25.4.2011. - 15:45
7. **http://www.softpanorama.org/Net/tcp\_protocol\_layers.**shtml - 25.4.2011. - 15:46
8. **http://www.freesoft.org/CIE/Course/Section4/index.htm** - 25.4.2011. - 15:47
9. **http://voip.neslanovac.com/voip-faq/81-tcp-transmission-control-protocol.html** - 25.4.2011. - 15:48
10. **http://mreze.layer-x.com/s040100-0.html** - 25.4.2011. - 15:49
11. **http://en.wikipedia.org/wiki/User\_Datagram\_Protocol** - 25.4.2011. - 15:51
12. **http://voip.neslanovac.com/voip-faq/78-udp-user-datagram-protokol.html** - 25.4.2011. - 15:52
13. **http://www.windowsnetworking.com/articles\_tutorials/Understanding-UDP-Protocol.html** - 25.4.2011. - 15:53
14. **http://searchsoa.techtarget.com/definition/UDP** - 25.4.2011. - 15:54
15. **http://compnetworking.about.com/od/networkprotocolsip/g/udp-user-datagram-protocol.htm** - 25.4.2011. - 15:54