

**Seminarski rad iz predmeta Računalne mreže
i usluge**

Naslov: Connectionless Transport: UDP

Studenti:

Tena Pavić

Mirela Stanić

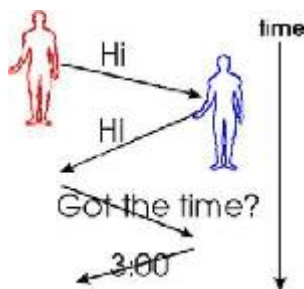
Vedrana Pandžić

Sadržaj

1	Uvod: Što je to protokol?	3
2	UDP- user datagram protokol	5
2.1	UDP vs. TCP	6
2.2	PREDNOSTI UDP-a	7
2.3	UDP-ova STRUKTURA SEGMENTA	10
2.4	UDP-ov CHECKSUM	10
3	Zaključak	12

1 Uvod: Što je to protokol?

Ljudi u svakodnevnom životu koriste protokole. Pogledajmo na primjeru (slika 1): ako želimo pitati nekoga koliko je sati. Prvo ćemo osobu pozdravit: "Dobar dan", te pričekati odgovor. Ako nam druga osoba uzvrati pozdrav, to možemo shvatiti kao odobrenje za nastavak razgovora i upitati koliko ima sati. U suprotnom, ako nam osoba ne uzvrati pozdrav ili nam kaže da ju ne smetamo, naš "ljudski" protokol nam govori da prekinemo razgovor i ne upitam ju za vrijeme. Očito je da poruke koje mi šaljemo, u ovom primjeru pozdrav, i primamo kao odgovor čine glavnu ulogu u našem "ljudskom" protokolu. Mrežni protokol je vrlo sličan "ljudskom" protokolu, samo što umjesto ljudi komuniciraju računala putem računalne mreže.



Slika 1:

Internet protokol je mrežni protokol za prijenos podataka kojeg koriste izvorišna i odredišna računala za uspostavu podatkovne komunikacije putem računalne mreže. Za korisnike i programere najvažnije je svojstvo Internet protokola (IP) da se ne razlikuje u primjeni za lokalne (LAN) i globalne (WAN) računalne mreže. To znači da program, pa ni čovjek, komunicirajući s drugim programima ili ljudima ne trebaju znati gdje se njihov sugovornik fizički nalazi, niti provoditi posebne postupke u ovisnosti o tome.

Internet protokol prilično je jednostavan. Podaci se razmjenjuju u paketima (slika 2). Paket se sastoji od zaglavlja i od podataka. Zaglavlje ima odredišnu adresu, izvorišnu adresu i nekoliko dodatnih atributa. Prostor s podacima sadrži podatke onako kako ih je korisnik želio poslati. Cijeli ovakav paket naziva se "Internet Datagram"¹. Specifično je da se prilikom slanja paketa između izvorišta i odredišta unaprijed ne određuje točan put

¹datagram- kratka poruka udp protokola

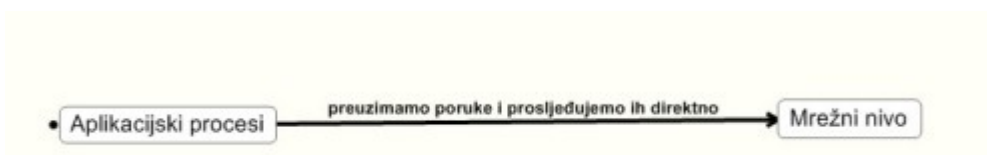
preko mreže kojim će podaci prijeći, te u tom smislu govorimo o IP mreži kao o "paketskoj" mreži. Ovaj osnovni prijenos zovemo "nepouzdanim". Razlog tome nije što on ne prenosi podatke pouzdano već u tome što sam protokol nema mehanizam potvrde prijena. Slično, kao kada šaljemo razglednicu: ne dobijemo potvrdu da je ona zaista stigla na cilj. Ako nam je ta potvrda važna, pošta nam ju ne može dati, već u sadržaju razglednice moramo zamoliti primatelja da on nama pošalje razglednicu na kojoj će pisati da je primio našu.



Slika 2:

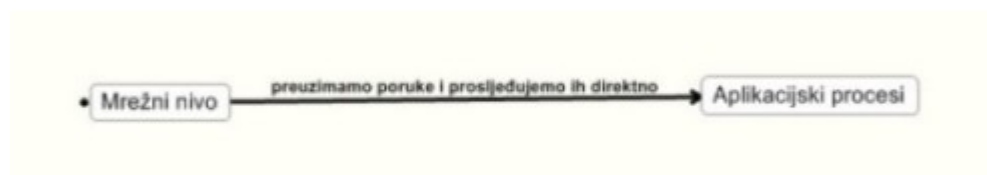
2 UDP- user datagram protokol

Kako bismo motivirali diskusiju o UDP-u i TCP-u kao dvama najvažnijim transportnim protokolima, pretpostavit ćemo da smo zainteresirani za dizajniranje najjednostavnijeg protokola. Prvo, u obzir moramo uzeti "prazan" transportni protokol. Gledajući sa strane pošiljalca imamo:



Slika 3:

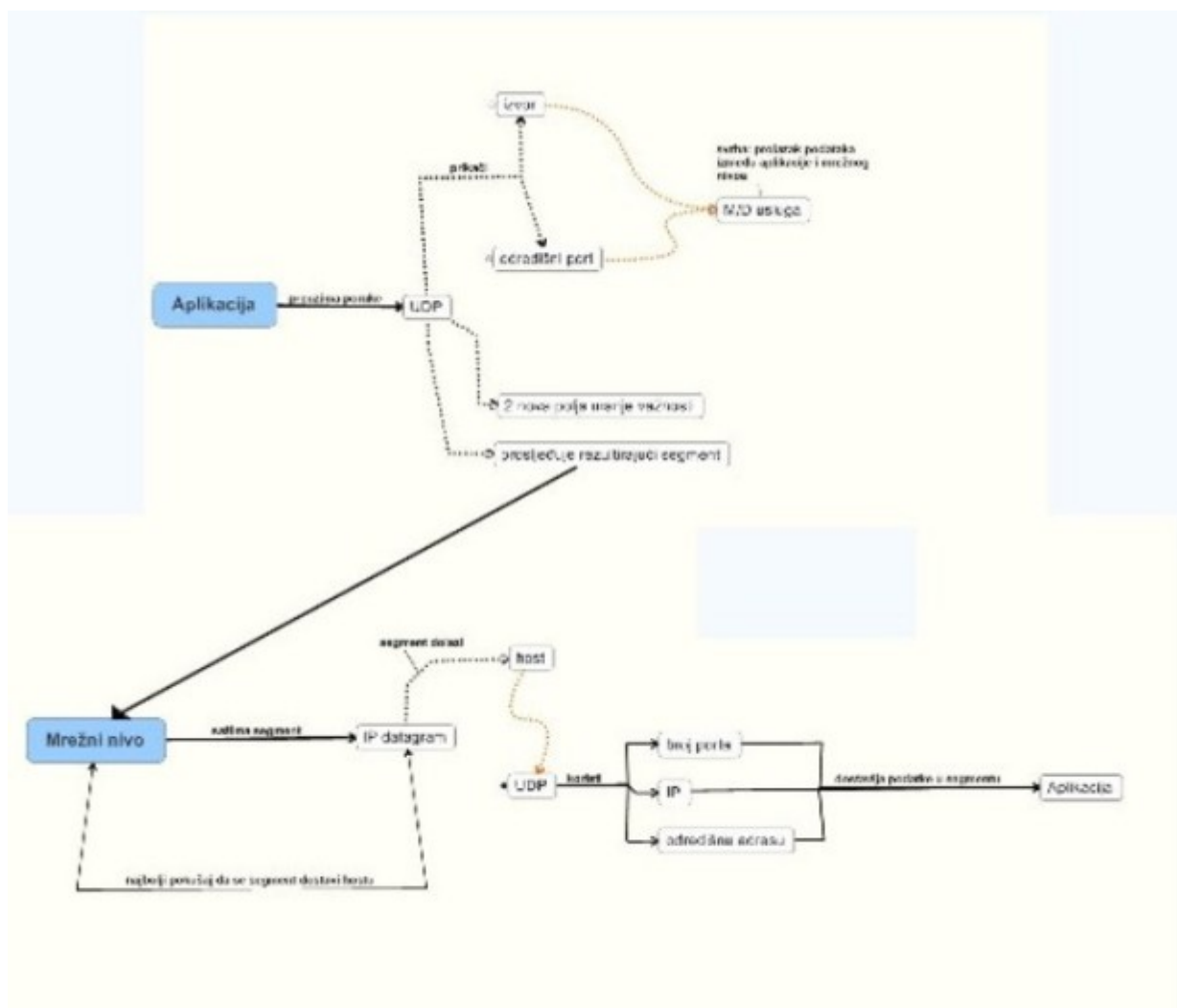
a s primateljske strane imamo:



Slika 4:

2.1 UDP vs. TCP

UDP omogućuje najjednostavnije funkcije koje nudi transportni protokol. Također omogućuje slanje datagrama između aplikacija. Uz multiplexing/demultiplexing²
³ UDP posjeduje i mogućnost provjeravanja pogreške prilikom prenošenja podataka, ali bez obavijesti o primitku poruke. Radi na sljedećem principu:



Slika 5:

² multiplexing- višestruko prenošenje signala

³ demultiplexing- razdvajanje signala

OPIS:

- UDP preuzima poruke od aplikacija;
- prikači izvor i odredišni port za multiplexing/demultiplexing uslugu;
- dodaje 2 nova polja manje važnosti;
- prosljeđuje rezultirajući segment do mrežnog nivoa;
- mrežni nivo sažima segment u IP datagram;
- ako segment dođe do host-a, UDP koristi broj porta, IP i odredišnu adresu da bi dostavio podatke u segmentu do odgovarajuće aplikacije

Kod UDP-a ne postoji direktan kontakt između primanja i slanja objekata. Zato se za UDP kaže i da je connectionless⁴.

TCP spada u grupu spojnih protokola, za razliku od UDP-a. Ovaj protokol je pouzdan, to znači da osigurava potvrdu prijema kao i ispravan redoslijed primljenih paketa i provjeru ispravnosti prenesenih podataka. I TCP koristi multiplexing kao i portove. Pruža i mogućnost konekcija prema jednoj aplikaciji na jednom hostu od strane više klijenata.

2.2 PREDNOSTI UDP-a

Iako se na prvi pogled čini da TCP zbog pouzdanog prijenosa podataka ima prednost u odnosu na UDP, postoje programi kojima više "odgovara" UDP. Navest ćemo nekoliko razloga u korist UDP-a:

- *Nema uspostave veze:*

Prije početka slanja podataka, TCP omogućuje uspostavu veze tj. konekcije između računala preko koje će slati podatke, dok UDP ne uspostavlja vezu. DNS⁵ ide preko UDP-a jer mu pouzdanost nije toliko važna kao brzina. S druge strane, pouzdanost je mnogo važnija za tekstualne web stranice pa se HTTP koristi TCP-om.

- *Ne čuva informaciju o stanju veze:*

Kod prijenosa podataka, TCP omogućava potvrdu prijema podataka, osigurava ispravan redoslijed primljenih paketa, te provjeru ispravnosti prenesenih podataka. Za razliku od TCP-a, UDP ne čuva informacije o stanju veze i radi na principu "pošalji i zaboravi".

- *Small segment header overhead*

⁴connectionless-bespojni. Bez izravne veze

⁵DNS - DOMAIN NAME SYSTEM (pretvara brojčanu IP adresu u simboličko ime)

TCP-ov segment ima 20 bajta zaglavlja overhead, dok UDP ima samo 8 bajta overheada.

- *Neregulirana brzina:*

TCP ima kontrolu zagušenosti kojom regulira pošiljatelja kada jedna ili više veza između pošiljatelja i primatelja postanu zagušene. Ta regulacija može imati značajan utjecaj na "real time" aplikacije koje mogu tolerirati mali gubitak podataka, ali zahtijevaju minimalnu brzinu prijenosa. Brzina pri kojoj UDP šalje podatke je ograničena samo brzinom kojom aplikacije generira podatke, kapacitetom izvorišnog računala i širinom prijenosa preko Interneta. Međutim, moramo uzeti u obzir da primatelj ne mora primiti sve podatke zbog zagušenja mreže, isto tako značajan dio UDP-ovih prijenosnih podataka se može izgubiti zbog prekoračenja na baferu. Dakle, brzina pri kojoj se primaju podaci je ograničena mrežnim zagušenjem čak i ako brzina pri kojoj šaljemo podatke nije.

U tablici(tablica 1) možemo vidjeti koje protokole koriste različite Internet aplikacije. Očekivano, e-mail(SMTP)⁶, Web(HTTP)⁷, prijenos podataka(FTP)⁸ te aplikacije za pristup udaljenom terminalu(TelNet)⁹ koriste TCP. Ipak, mnoge druge važne aplikacije idu preko UDP, kao što su: Internet telefonija, Network Managent(SNMP)¹⁰, RIP¹¹, DNS. Možemo zaključiti da se UDP više koristi za multimedijalne aplikacije: "real time" videokonferencije, Internet telefonija, prikazivanje audio i video zapisa. Sve ove aplikacije mogu tolerirati mali gubitak podataka tijekom slanja, tako da pouzdani prijenos podataka nije od ključne važnosti za "uspjeh" ovih aplikacija. Isto tako aplikacije kao što su videokonferencije i Internet telefon reagiraju jako slabo na TCP-ovu kontrolu zagušenosti. Zbog tih se razloga programeri multimedijalnih aplikacija koriste UDP-om. Također, "multicast" aplikacije, tj. aplikacije koje omogućavaju slanje podataka prema većem broju korisnika, koriste UDP jer TCP ne podržava "multicasting".

No međutim postoji potencijalno veliki problem kod UDP-a. Ako bi npr.

⁶SMTP- SIMPLE MAIL TRANSFER PROTOCOL (protokol za razmjenu elektroničke pošte)

⁷HTTP- HYPERTEXT TRANSFER PROTOCOL (protokol za pregled web stranica)

⁸FTP- FILE TRANSFER PROTOCOL (protokol za prijenos datoteke)

⁹TELNET-TELEphone NETwork (protokol dizajniran s namjerom povezivanja jednog terminala na udaljeno računalo)

¹⁰SNMP- SIMPLE NETWORK MANAGMENT PROTOCOL

¹¹RIP- ROUTING INFORMATION PROTOCOL (unutrašnji protokol za usmjerenje)

veliki broj korisnika istodobno počeo "skidati" neki video isječak, moglo bi doći do zagušenja mreže, prekoračenja podataka na ruterima, a korisnicima ne bi bilo omogućeno gledanje tog video isječka. Veliki nedostatak UDP-a je upravo to što nije u mogućnosti kontrolirat tu zagušenost.

Kako je već spomenuto, UDP ne čuva informacije o stanju veze, a ipak je moguće da ostvariti pouzdani prijenos podataka. Za to su zaslužne aplikacije, jer je ta "pouzdanost" ugrađena u njih same. Naime, programeri su omogućili aplikacijama da same sebi "isporučuju" potvrdu o ispravnom prijenosu podataka, te da u slučaju neispravnog prijema, ponovno pošalju podatke. Tako unaprijeđene aplikacije koje idu preko UDP-a ulaze su u svakodnevnu uporabu.

Aplikacija	Protokol aplikacijskog sloja	Donji transportni protokol
elektronička pošta	SMTP	TCP
daljinski uređaj za pristup	Telnet	TCP
web	HTTP	TCP
prijenos podatak	FTP	TCP
usluga udaljenog servera	NFS	UDP
streaming multimedia	protokol proizvođača	UDP
internet telefonija	protokol proizvođača	UDP
upravljanje mrežom	SNMP	UDP
kontrola usmjeravanja	RIP	UDP
prevođenje imena	DNS	UDP

Slika 6:

2.3 UDP-ova STRUKTURA SEGMENTA

Podaci aplikacije zauzimaju podatkovna polja UDP-ovog datagrama (tablica 2). Na primjer za DNS podatkovna polja sadrže ili upitnu poruku ili odgovor. Za audio aplikaciju, audio primjerak ispunjava podatkovno polje. UDP-ovo zaglavlje ima samo četiri polja od kojih se svaki sastoji od četiri byte-a. To su: izvorišni i odredišni port, duljina i checksum¹². Broj porta dopušta primatelju prijenos podataka aplikacije do odgovarajućeg procesa koji se obavlja na tom primatelju (npr. obavlja demultiplexing). Primatelj koristi checksum da bi provjerio ima li pogrešaka u segmentu koje su se mogle dogoditi prilikom slanja podataka od izvora do odredišta.



Slika 7:

2.4 UDP-ov CHECKSUM

UDP-ov checksum omogućava otkrivanje pogreške prilikom slanja podataka. Na primateljskoj strani UDP izvodi primarni komplement sume svih 16-bitnih riječi u segmentu. Ovaj rezultat se stavlja u polje checksum-a UDP-ova segmenta. Kada segment stigne (ako stigne) do primatelja, sve 16-bitne riječi se zbrajaju zajedno, uključujući i checksum. Ako je suma jednaka 1111111111111111, onda segment nema pogrešaka. Ako je jedan od bitova 0, onda znamo da se greška pojavljuje u segmentu.

Ovdje je jednostavan primjer checksum izračuna. Za primjer uzimamo da imamo ove tri 16-bitne riječi:

```
0110011001100110
0101010101010101
0000111100001111
```

Suma prve dvije riječi je:

¹²Checksum - Zbroj za provjeru

0110011001100110
0101010101010101

1011101110111011

Dodajemo treću i imamo:

1011101110111011
0000111100001111

1100101011001010

Primarni komplement nastaje pretvaranjem svih 0 u 1 i svih 1 u 0. Tako da je prvi komplement sume od 1100101011001010 jednak 0011010100110101, koja postaje checksum. Kod primatelja, sve četiri 16-bitne riječi se zbrajaju, uključujući i checksum. Ako nema pogrešaka u segmentu, jasno je da će suma biti 1111111111111111. Checksum nije potpuno pouzdan - čak ako je suma jednaka 1111111111111111 i dalje je moguće da se jave neotkrivene pogreške u segmentu. Iz ovog razloga, velik broj protokola koristi mnogo sofisticiranije metode detekcije grešaka, mjesto jednostavnog checksum-iranja.

Zašto UDP omogućava checksum uopće, kao što mnogi protokoli na nivou povezivanja također omogućuju provjeru grešaka?

Razlog je to nema garancije da sve veze između izvora i odredišta omogućuju provjeru pogrešaka - jedna od veza može upotrebljavati protokol koje ne omogućuje provjeru grešaka. Iz razloga to IP ide preko bilo kojeg protokola, korisno je za transportne nivoe da omogućuje provjeru grešaka kako sigurnosne mjere. Iako UDP omogućava provjeru grešaka, on ne čini ništa kako bi istu grešku ispravio. Neke od implementacija UDP-a jednostavno odbacuju oštećene segmente, dok drugi pređu preko oštećenog segmenta s upozorenjem.

3 Zaključak

I UDP i TCP imaju svojih prednosti i nedostataka. UDP se koristi kada je važnija brzina i efikasnost od pouzdanosti kao kod npr. multimedijalnih aplikacija. S druge strane, TCP je po prirodi složeniji od UDP, pa samim time omogućuje pouzdanu i u kontroliranom redoslijedu isporuku podataka od pošiljatelja do primatelja. No napretkom tehnologije(programerstva) postoje mogućnosti unaprjeđivanja UDP-a, pa tako i UDP postaje pouzdan protokol za prijenos podataka.