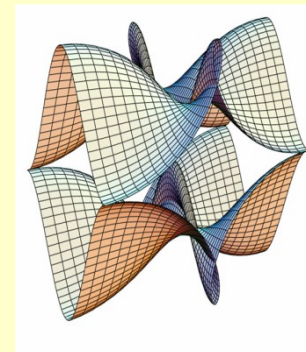




Sveučilište u Zagrebu  
PMF – Matematički odsjek

MREŽE RAČUNALA  
Predavanja 2022/2023



# Poglavlje 18: Jednostavni transportni protokol – UDP

Sastavili: Luka Grubišić i Robert Manger  
08.12.2014

# Uvod

- U prethodnom predavanju prikazali smo bezspojnu paradigmu za slanje poruka o grešci u IP komunikaciji.
- U ovom predavanju prikazat ćemo bezspojnu komunikacijsku paradigmu koja omogućava proizvoljnim **aplikacijama** izmjenu poruka kroz heterogenu mrežu.
- Koristi se semantika traženja najboljeg rješenja (*best effort communication*), koja ima neke neželjene posljedice.

# End-To-End transportni protokol

- Protokol koji omogućava izmjenu podataka između dvije aplikacije nazivamo *End-To-End* transportni protokol.
- IP ne pruža takvu uslugu budući da IP zaglavlje identificira računala kao krajnje točke u komunikaciji.
- U TCP/IP stogu protokola *End-To-End* transportni protokoli se nalaze u sloju 4.
- Operativni sustav host računala pruža dodatnu uslugu prosljeđivanja poruke do krajnjeg odredišta, tj. aplikacije.

# UDP protokol (1)

- U TCP/IP stogu protokola User Datagram Protokol UDP pruža uslugu bezspojne *End-To-End* komunikacije.
- UDP komunikacija se može karakterizirati kao:
  - End-To-End: UDP protokol može identificirati ciljnu aplikaciju na danom računalu.
  - Bezspojna: Sučelje koje UDP pruža aplikacijama implementira paradigmu bezspojne komunikacije.
  - Orijentirana na slanje poruka: Osnovna komunikacijska jedinica u UDP protokolu se naziva poruka.

# UDP protokol (2)

- Best-effort semantika: UDP pruža uslugu koja koristi *best-effort* semantiku IP protokola.
- Bez ograničenja na broj interakcija: UDP dopušta aplikaciji slanje poruka prema neograničenom broju drugih aplikacija, te primanje poruka od neograničenog broja drugih aplikacija ili komunikaciju s točno jednom aplikacijom.
- Neovisnost o operativnom sustavu: UDP pruža mehanizam za identifikaciju ciljne aplikacije. Mehanizam je neovisan o operativnom sustavu host računala.

# Bezspojna usluga

- Bezspojna usluga znači da aplikacija koja koristi UDP protokol ne mora prije slanja podataka uspostaviti komunikacijski kanal s ciljnom aplikacijom.
- UDP ne koristi kontrolne poruke. Posebno:
  - UDP dopušta proizvoljno dugo čekanje između slanja dviju poruka.
  - Ukoliko dvije aplikacije prestanu komunicirati, nikakve daljnje poruke o tome se ne šalju.
- Osnovna značajka UDP protokola je efikasnost.

# Sučelje za prosljeđivanje poruka (1)

- UDP pruža aplikacijama sučelje za slanje poruka.
- UDP ne dijeli poruke u pakete i ne sastavlja poruke po primitku.
- Svaka poruka koju aplikacija pošalje direktno se prenosi kao zasebni IP datagram kroz Internet do krajnjeg odredišta.
- Rezultat je:
  - **Pozitivno:** Programer se kod slanja poruke može osloniti na protokol i ne mora razmišljati o načinu kako se ta poruka prenosi kroz mrežu.
  - **Negativno:** Poruke se ne fragmentiraju, pa je veličina IP datagrama gornja ograda za veličinu UDP poruke.

# Sučelje za prosljeđivanje poruka (2)

- Rezultat je da UDP poruke mogu uzrokovati neefikasno korištenje fizičke mreže.
- Ukoliko aplikacija šalje male UDP poruke omjer korisnog tereta i zaglavlja će biti loš. Protokol takvo korištenje mreže ne ograničava.
- Ukoliko aplikacija šalje jako velike poruke, UDP protokol ne osigurava da datagrami koji se šalju neće biti veći od MTU-a mreže.
- Anomalija UDP protokola: **Slanje (pre)velikih poruka usporava komunikaciju.** Ponekad UDP šalje poruke koje **IP protokol mora fragmentirati već na polaznom računalu.**



# Semantika UDP komunikacije (1)

- UDP koristi IP protokol za svu komunikaciju.
- Aplikacijama pruža istu (naslijeđenu) semantiku *best-effort* komunikacije koju ima i IP protokol.
- Zbog toga je naslijeđen i problem gubitka poruke, dupliciranja poruke, te gubitak redosljeda slanja poruka.
- UDP ne identificira i ne rješava probleme koji nastaju zbog ovih svojstava IP protokola.
- UDP je pogodan za aplikacije koje šalju audio ili video podatke. Nije pogodan za aplikacije gdje je redosljed događaja bitan.

# Semantika UDP komunikacije (2)

- UDP dopušta sljedeće komunikacijske paradigme
  - 1 prema 1
  - 1 prema mnogo
  - mnogo prema 1
  - mnogo prema mnogo
- Mnogo prema mnogo paradigma znači da grupa aplikacija može grupno izmijenjivati poruke.
- Drugim riječima ...

# Unicast, Multicast, Broadcast

- Paradigma 1 prema mnogo bi se mogla riješiti slanjem pojedinačnog paketa svakom primatelju.
- Ali UDP optimizira ovakvu komunikaciju korištenjem IP *multicast* ili *broadcast* adresa. Npr. lokalni *broadcast* može biti riješen slanjem na adresu 255.255.255.255.
- Takav način komunikacije je posebno pogodan za korištenje na *Ethernet* mrežama.

# Interakcija s operativnim sustavom

- UDP ne može koristiti isti mehanizam koji koristi i operativni sustav za identifikaciju pojedinačnih aplikacija.
- Različita računala koriste različite mehanizme: *process number, job number, task identifier ...*
- UDP definira apstraktni niz identifikatora za procese zvan *protocol port numbers* ili neformalno *port*.
- *Protocol port numbers* su neovisni o operativnom sustavu.
- Na računalima na kojima je implementiran UDP, operativni sustav pruža uslugu povezivanja procesa i port-ova.

# Interakcija s operativnim sustavom (2)

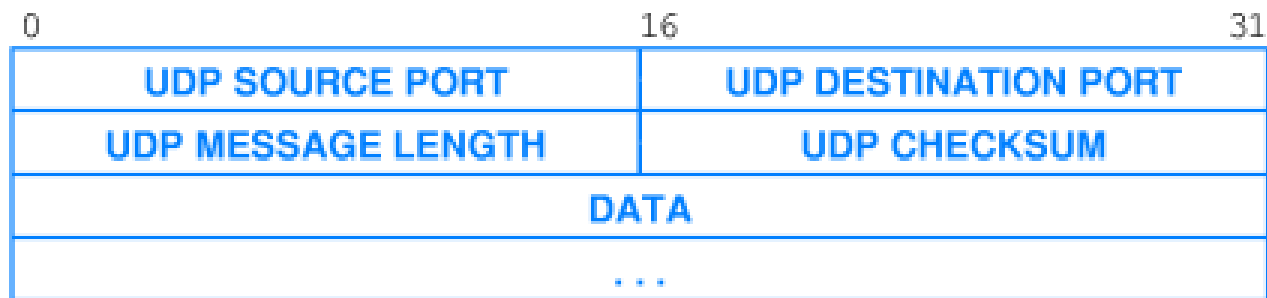
- UDP standard određuje da **port 7** pripada aplikaciji **echo**, dok **port 37** pripada aplikaciji **timeserver**.
- Sva računala na kojima se izvršava UDP prepoznaju standardne port-ove neovisno o operacijskom sustavu. Ako dođe poruka na **port 7**, operativni sustav mora proslijediti poruku onom procesu u kojem se izvršava **echo** servis.

# Interakcija s operativnim sustavom (3)

- Za realizaciju 1-1 komunikacije po UDP protokolu aplikacija se veže na lokalni port i specificira IP adresu (i port) udaljenog računala. UDP tada prosljeđuje samo one poruke koje stižu sa specificirane IP adrese na dani port.
- Za realizaciju mnogi-1 paradigme po UDP protokolu aplikacija određuje lokalni port i informira UDP da pošiljatelj može biti proizvoljan. UDP tada prosljeđuje aplikaciji sve poruke koje dođu na dani port.
- UDP dopušta samo jednoj aplikaciji da se u danom trenutku veže na dani lokalni port.

# Format UDP datagrama

- UDP poruka se naziva *user datagram*.
- Sastoji se od dva dijela
  - zaglavlja koje određuje polaznu i krajnju aplikaciju,
  - korisnog tereta, gdje se nalaze podaci koji se šalju.



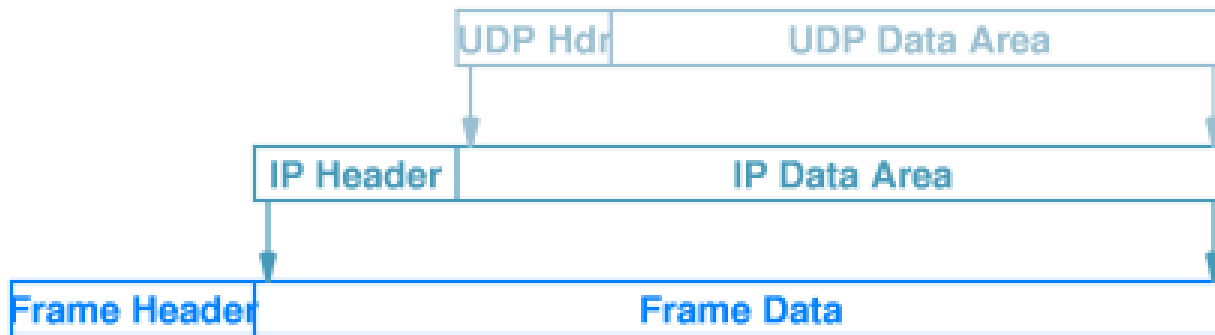
# UDP pseudo zaglavlje

- UDP zaglavlje sadrži polje od 16 bitova koje se zove UDP CHECKSUM. Korištenje je opcionalno (ako se ne koristi, postavi se na 0).
- UDP pretpostavlja da se IP adrese pošiljatelja i primatelja nalaza u IP datagramu. Zbog toga u zaglavlju stoje samo brojevi port-ova.
- Ova optimizacija smanjuje količinu dodatnih podataka koji se prenose, ali povećava mogućnost greške u transmisiji.
- Na primjer, UDP ne bi mogao detektirati slučaj kad je IP proslijedio poruku na krivu adresu.
- Da bi se spriječila prethodna greška, UDP-ova kontrolna suma pokriva i IP adrese pošiljatelja i primatelja i IP TYPE. To je tzv. pseudo-zaglavlje.



# UDP enkapsulacija

- UDP koristi IP za transport.
- Enkapsulacija se provodi slično kao i kod ICMP protokola.



# Sažetak

- UDP protokol pruža aplikacijama uslugu *End-To-End* slanja poruka.
- UDP pruža istu semantiku *best-effort* komunikacije kao i IP protokol.
- UDP poruke se enkapsuliraju u IP datagrame.
- UDP zaglavlje ne sadrži IP adrese pošiljatelja i primatelja.
- Pored grešaka koje su moguće unutar IP protokola, moguće je da poruke koje imaju kontrolnu sumu postavljenu na 0 stignu na krivo računalo.
- Korištenjem pojma *protocol port number* UDP je neovisan o operativnom sustavu host-a.