

TEHNIČKA ŠKOLA RUĐERA BOŠKOVIĆA
Zagreb, Getaldićeva 4

Niko Josipović, III-b
Protokoli transportnog sloja (TCP i UDP)
LABORATORIJSKA VJEŽBA

Zagreb, ožujak 2025.

PRIPREMA ZA VJEŽBU

1. Koje su prednosti i nedostaci protokola TCP?

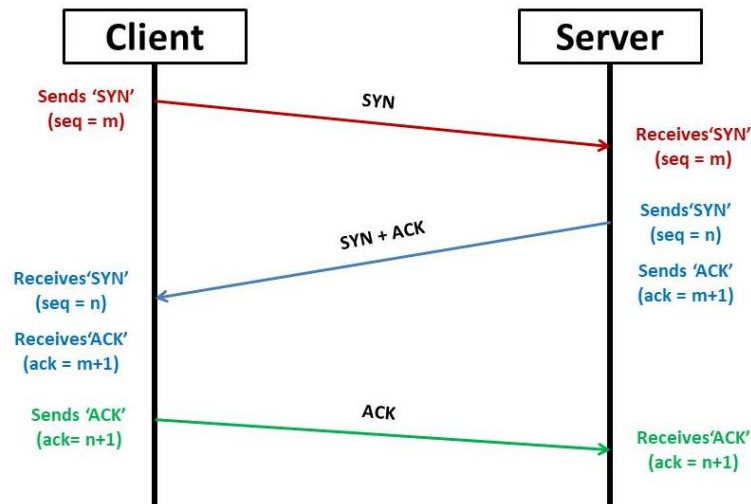
PREDNOSTI	NEDOSTATCI
Pouzdan prijenos podataka uz potvrdu isporuke	Veća potrošnja resursa (CPU, memorija, širina pojasa)
Kontrola toka i korekcija grešaka	Sporiji od UDP-a zbog uspostave i raskida veze
Osigurava ispravan redoslijed paketa	Nije pogodan za aplikacije u stvarnom vremenu (kašnjenje)
Automatska retransmisija izgubljenih paketa	Veća složenost implementacije
Prikladan za prijenos važnih podataka (HTTP, e-mail, FTP)	Ne podržava multicast i broadcast prijenos

2. Koje su prednosti i nedostaci protokola UDP?

PREDNOSTI	NEDOSTATCI
Brža komunikacija zbog izostanka uspostave veze	Nema mehanizma za potvrdu isporuke podataka
Manja potrošnja resursa (CPU, memorija)	Paketi mogu stići neurednim redoslijedom ili biti izgubljeni
Pogodan za aplikacije u stvarnom vremenu (VoIP, streaming, online igre)	Nema kontrole toka podataka
Omogućuje multicast i broadcast prijenos	Nema ugrađene enkripcije ni sigurnosnih mehanizama
Jednostavnija implementacija u odnosu na TCP	Potrebna dodatna obrada na razini aplikacije za pouzdan prijenos

3. Skiciraj i objasni postupak uspostave TCP veze između klijenta i poslužitelja.

Kad TCP hosta pošiljatelja uspostavlja vezu, šalje segment naziva SYN ravnopravnom TCP protokolu na hostu primatelju. TCP primatelja vraća segment nazvan ACK kako bi potvrdio uspješan prijam segmenta. TCP pošiljatelj šalje drugi ACK segment i zatim nastavlja slati podatke. Ova razmjena kontrolnih informacija naziva se trostrukim rukovanjem.



ODGOVORI NA PITANJA

1. Što je TCP?

TCP je mrežni protokol koji kod komunikacije dvaju računala provjerava da je svaki segment podataka koji je poslan komunikacijskim kanalom od pošiljatelja do primatelja primljeno u ispravnom obliku.

2. Koji protokoli aplikacijske razine koriste TCP?

Protokoli aplikacijske razine koji koriste TCP uključuju:

- **HTTP/HTTPS** (za web stranice)
- **FTP** (za prijenos datoteka)
- **SMTP, IMAP, POP3** (e-mail komunikacija)
- **SSH** (sigurna daljinska prijava)
- **Telnet** – daljinski pristup uređajima
- **LDAP** – pristup direktorijskim servisima
- **SMB** – dijeljenje datoteka u mrežama
- **MQTT** – komunikacija u IoT sustavima

3. Navedite dvije osnovne karakteristike TCP protokola?

Pouzdanost: osigurava isporuku podataka bez grešaka, u ispravnom redoslijedu, uz potvrdu primitka i retransmisiju izgubljenih paketa.

Kontrola toka i zagušenja: regulira količinu podataka koji se šalju kako bi se izbjeglo preopterećenje mreže i prijemnika.

4. Koje su faze procesa rukovanja kod TCP protokola? Opišite i nacrtajte dijagram.

Uspostava veze (Three-Way Handshake)

1. **SYN** - klijent šalje zahtjev za uspostavu veze s oznakom **SYN** (*synchronize*)
2. **SYN-ACK** - poslužitelj odgovara s **SYN-ACK** (potvrda primanja zahtjeva i slanje vlastitog zahtjeva)
3. **ACK** - klijent šalje završnu potvrdu **ACK** (*acknowledgment*), veza je uspostavljena

5. Što rade klijent i server tijekom postupka rukovanja? Što je ISN?

Klijent i server tijekom TCP rukovanja:

- **Klijent** započinje komunikaciju slanjem **SYN** paketa s početnim sekv. brojem (**ISN**)
- **Server** odgovara **SYN-ACK** paketom, potvrđuje primljeni ISN klijenta i šalje vlastiti
- **Klijent** potvrđuje **ACK** paketom, čime je veza uspostavljena

ISN (Initial Sequence Number) je nasumično generirani početni sekvencijski broj koji se koristi za identifikaciju prvog paketa u TCP vezi. Omogućuje sigurnost i pravilno slaganje paketa u redoslijed.

6. Objasnite generičke TCP parametre paketa.

Parametar	Opis
Izvorni port	Nema mehanizma za potvrdu isporuke podataka
Odredišni port	Paketi mogu stići neurednim redoslijedom ili biti izgubljeni
Sekvencijski broj	Nema kontrole toka podataka
Potvrđni broj	Nema ugrađene enkripcije ni sigurnosnih mehanizama
Zastavice (Flags)	Potrebna dodatna obrada na razini aplikacije za pouzdan prijenos
Prozor	Definira veličinu primljenih podataka prije potvrde
Kontrolna suma	Provjerava ispravnost podataka
Opcije	Dodatne TCP konfiguracije (npr. maksimalna veličina segmenta - MSS)

7. Objasnite bar tri TCP zastavice.

- **SYN** - inicira uspostavu veze
- **ACK** - potvrđuje primitak podataka ili zahtjeva
- **FIN** (*Finish*) - Signalizira zatvaranje veze

IZVOĐENJE VJEŽBE

1. Analizirati zaglavlje odlaznih i dolaznih TCP segmenata

- a) Pronaći segmente pomoću kojih se uspostavila veza između klijenta i poslužitelja (SYN, SYN-ACK, ACK)

3	1.088179	192.168.123.14	161.53.160.228	TCP	66	50008 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
4	1.088351	192.168.123.14	161.53.160.228	TCP	66	50009 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
7	1.089403	161.53.160.228	192.168.123.14	TCP	66	80 → 50008 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM WS=128
8	1.089431	192.168.123.14	161.53.160.228	TCP	54	50008 → 80 [ACK] Seq=1 Ack=1 Win=262656 Len=0

Označeni SYN segment ne pripada otvaranju veze u TCP protokolu.

- b) Pronađene segmente usporedite sa skicom iz pripreme, zadatak 3.

Identično je

- c) Koji je broj ishodišnog priključka (engl.port)?

50008

- d) Koji je broj odredišnog priključka (engl.port)?

80

- e) Pronađite brojeve koji označavaju redni broj segmenata (SEQ) i komentirajte!

```
66 50008 → 80 [SYN] Seq=0 Win=
66 50009 → 80 [SYN] Seq=0 Win=
66 80 → 50008 [SYN, ACK] Seq=0
54 50008 → 80 [ACK] Seq=1 Ack=
```

po redu:

```
Sequence Number: 0 (relative sequence number)
Sequence Number (raw): 1757923270
```

```
Sequence Number: 0 (relative sequence number)
Sequence Number (raw): 3529293214
```

```
Sequence Number: 1 (relative sequence number)
Sequence Number (raw): 2634765914
```

redni broj: Wireshark-ov čitljiviji prikaz rednog broja. Izračunava se počevši od 0, tako da je lakše pratiti pakete.

Broj segmenta SYN (koji mi šaljemo) jednak je 0, stoga je broj segmenta naknadno poslanog ACK segmenta jednak 1. SYN, ACK ima -> seq = 0 Zato što taj segment šalje poslužitelj.

redni broj (neobrađeni): stvarni redni broj poslan u paketu - on počinje od ISN-a

- f) Čemu služi oznaka Win?

Oznaka Win odnosi se na Window Size ili veličinu TCP prozora.

Ovo je veličina prijemnog međuspremnika u trenutnom hostu koji odašilje. Host ovdje obavještava host druge strane koliko bajtova može primiti kako bi se izbjegao slučaj da druga strana odgovori s velikim brojem bajtova s kojima se ne može rukovati.

- g) Pronađite brojeve koji označavaju potvrdu primljenog segmenta (ACK) i komentirajte.

ACK broj u trećem paketu je serverov broj sekvence + 1 (tako potvrđuje primitak serverovog SYN-ACK paketa)

```
66 80 → 50008 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0
54 50008 → 80 [ACK] Seq=1 Ack=1 Win=262656 Len=0
```

h) Koja su ostala polja TCP zaglavlja? Istražite i zapišite čemu služe!

- **Dužina TCP segmenta:** Veličina podataka sadržanih u ovom paketu
- **Sljedeći redni broj:** Obično je to trenutni redni broj + duljina podataka u trenutnom paketu. Ovo se pravilo ne odnosi na ovaj paket jer je to SYN paket, a SYN se smatra 1 bajtom, tako da se sljedeći redni broj povećava za 1 sedam ako u paketu nema podataka.
- **Broj potvrde:** Ovo predstavlja ukupan broj bajtova koje je host trenutnog odašiljača primio od druge strane. Polje je dodao Wireshark kako bi se olakšalo analiziranje TCP snimanja brojanjem broja potvrde od 0.
- **Broj potvrde (neobrađeni):** pravi broj potvrde.
- **Dužina zaglavlja:** duljina TCP zaglavlja. To može biti u rasponu od 20 do 60 bajtova, ovisno o TCP opcijama u paketu.
- **Vrijednost veličine prozora:** Ovo je veličina prijemnog međuspremnika u trenutnom hostu koji odašilje. Host ovdje obavještava host druge strane koliko bajtova može primiti kako bi se izbjegao slučaj da druga strana odgovori s velikim brojem bajtova s kojima se ne može rukovati.
- **Kontrolni zbroj:** Kontrolni zbroj TCP paketa. Računalo primatelj ovo koristi za provjeru je li primljeni paket u redu
- **Status kontrolne sume:** prema zadanim postavkama Wireshark ne provjerava kontrolnu sumu paketa, ali postoji opcija za omogućavanje provjere kontrolne sume.

2. Analizirati zaglavlje odlaznih i dolaznih UDP segmenata

a) Pronaći UDP segmente

```
6 1.088750 192.168.123.14 162.159.61.3 UDP 307 52067 → 443 Len=265
12 1.090224 162.159.61.3 192.168.123.14 UDP 65 443 → 52067 Len=23
```

b) Koje protokole enkapsulira UDP?

NFS, SNMP, DNS, TFTP

c) Koji je broj ishodišnog priključka (engl.port)?

52067 - promatrajući 1. UDP segment

d) Koji je broj odredišnog priključka (engl.port)?

443 - promatrajući 1. UDP segment

e) Koja su ostala polja UDP zaglavlja? Istražite i zapišite čemu služe!

Polje **duljine podataka** (16 bitova) određuje duljinu zaglavlja i podataka.

Polje **kontrolnog zbroja** (16 bitova) pokazuje je li zaglavlje oštećeno tijekom transporta.

3. Koja je uloga priključka u TCP i UDP segmentima?

Uloga priključka je **omogućiti** uspostavu komunikacije između većeg broja aplikacija putem zajedničkog mrežnog sloja od programa na hostu izvoru podataka do programa na hostu odredišta.

4. Za poznate protokole koje ste „ulovili“ navedite predefinirane brojeve priključaka (za TCP ili UDP)

80 port -> HTTP

443 port -> HTTPS