УНИВЕРЗИТЕТ У НОВОМ САДУ
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА

УНИВЕРЗИТЕТ У НОВОМ САДУ
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА
НОВИ САД
Департман за рачунарство и аутоматику
Одсек за рачунарску технику и рачунарске комуникације

ИСПИТНИ РАД

Кандидат: **Кристина Пеце**
Број индекса: **РА 153/2016**

Предмет: **Међурачунарске комуникације и рачунарске мреже 1**
Тема рада: **Fast downloader client/dual-stack TCP server zasnovana arhitektura – Monoalfabetska enkripcija/dekripcija**

Ментор рада: **Проф. Илија Башичевић**

Нови Сад, Децембар 2022.

SADRŽAJ

1. Zadatak

IPv6 – Fast downloader client/dual-stack TCP server zasnovana arhitektura – Monoalfabetska enkripcija/dekripcija pri slanju i prijemu.

Po ugledu na programe za brzi prenos datoteka (engl. downloaders, GetRight, FlashGet, GoZilla), realizovati aplikacije IPv4 klijenta, IPv6 klijenta i *dual stack* servera (prima poruke generisane i preko IPv4 i preko IPv6 protokola) za prenos segmenta datoteka (od zadate pozicije, u zadatoj dužini) koristeći TCP protokol. Klijent po jednoj vezi prenosi deo datoteke. Formiranjem više istovremenih veza prenosi se cela datoteka, i ubrzava se proces prenosa. Server mora da simulira ograničenje brzine po vezi. Klijent po prijemu svih delova datoteke sklapa kompletnu datoteku. Porediti brzinu prenosa pomoću jedne i više veza – grafički prikazati rezultate analize u dokumentaciji. Server istovremeno opslužuje više klijenata.

Pri slanju podataka klijent enkriptuje sadržaj paketa Monoalfabetskom šifrom. Pri prijemu server vrši odgovarajuću dekripciju. I obrnuto. Ključn odrediti samostalno. Originalna poruka koja se enkriptuje (tj. datoteka) se sastoji samo od slova engleskog alfabeta. Napisati odgovarajuću dokumentaciju po ugledu na priloženi šablon.

2. Koncept rešenja

Rešenje zadatka se sastoji od sledećih koraka:

- **Realizacija *dual stack* servera** koji istovremeno može da prima poruke generisane i preko IPv4 i preko IPv6 protokola. To je moguće na dva načina.

  Jedna mogućnost je da na serveru se kreiraju dve uticnice, jedna za IPv4, a druga za IPv6 adresnu familiju. Zatim se popune dve adresne strukture sa podacima server: jedna tipa *sockaddr_in* za IPv4, a druga *sockaddr_in6* za IPv6. Povezivanje soketa sa adresama se obavi tako što bind-ujemo IPv4 soket sa IPv4 adresom, a IPv6 soket sa IPv6 adresom. Potrebno je još oba soketa staviti u neblokirajuci režim, i koristiti neki model neblokiranja (pooling ili select) da bi server mogao da prima poruke istovremeno sa obe vrste klijenata.

  Druga mogućnost je korišćenje serverske uticnice koja je kreirana za IPv6 adresnu familiju po osnovnom ponašanju šalje i prima iskljucivo IPv6 pakete. Za takvu uticnicu možemo dodatno postaviti opciju tako da šalje i prima i IPv4 i IPv6 pakete. Ova dodatna opcija se postavlja pomoću funkcije *setsockopt()*.

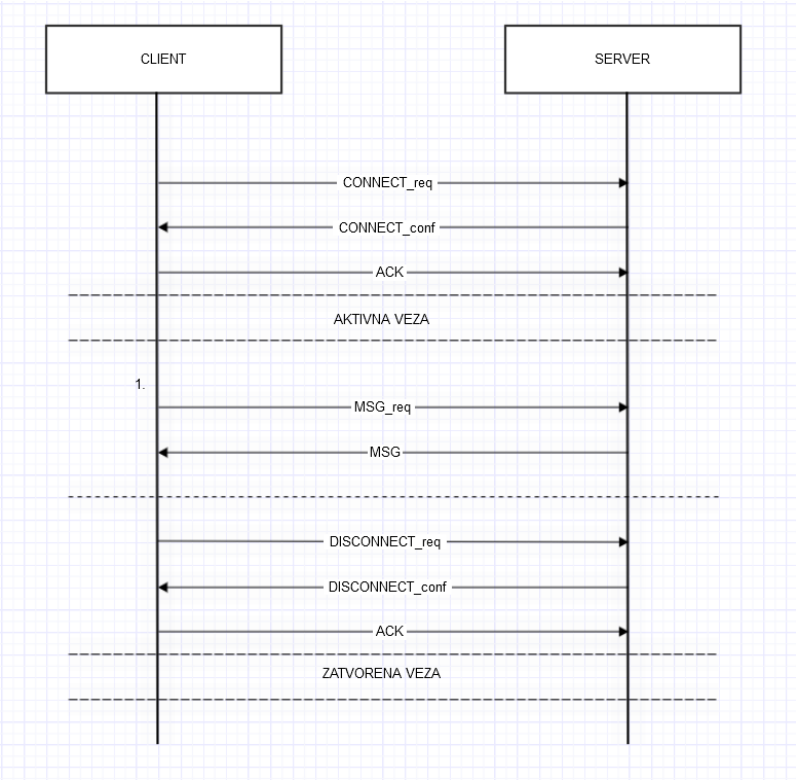
- **Uspostavljenje klijent/server veze koristeći TCP protokol** (Transmisioni kontrolni protokol) preko kojeg se realizuje prenos segmenta datoteka. Komunikacija uz pomoć TCP protokola se odvija tako što se prvo između klijenta i servera uspostavi veza (usluga sa konekcijom), za razliku od komunikacije UDP protokolom koji je ne zahteva. Konekcija se uspostavlja tako što se između prijemne i predajne strane iz tri puta razmene poruke sa podešenim odgovarajućim kontrolnim bitima.

- Realizacija servera koji je u mogućnost da uspostavi **više takvih veza** sa istim klijentom, tako što server otvori više utičnica. Nakon uspostavljenje prvobitne veze između klijenta i servera, klijent šalje poruku serveru u kojoj obaveštava preko koliko istovremenih veza će podatak biti poslat. Na ovo server odgovara sa potrebnim informacijama za uspostavljenje tih veza. Klijent po jednoj vezi prenosi deo datoteke. Datoteka se deli na toliko delova koliko istovremenih veza ima.

- Spajanje dobijenih segmenta u celinu. Ideja je da se segmenti imenuju tako što sadrži i redni broj spajanja.
- **Enkripcija/dekripcija** sadržaja paketa Monoalfabetskom šifrom. To je tip šifre proste zamene u kome se svako slovo originalne poruke menja odgovarajućim slovom alfabeta, pomerenim za određeni broj mesta koji definiše ključ. Na primer, sa pomakom 3, A se zamenjuje slovom G, B sa D itd.

3. Opis rešenja

Uzevši u obzir da je server dužan da kreria više utičnica, *dual stack* server je realizovan tako što je svaka utičnica sa strane servera podešen tako da šalje i prima i IPv4 i IPv6 pakete. Pomoću funkcije *setsockopt()* IPV6_V6ONLY se postavlja na 0.



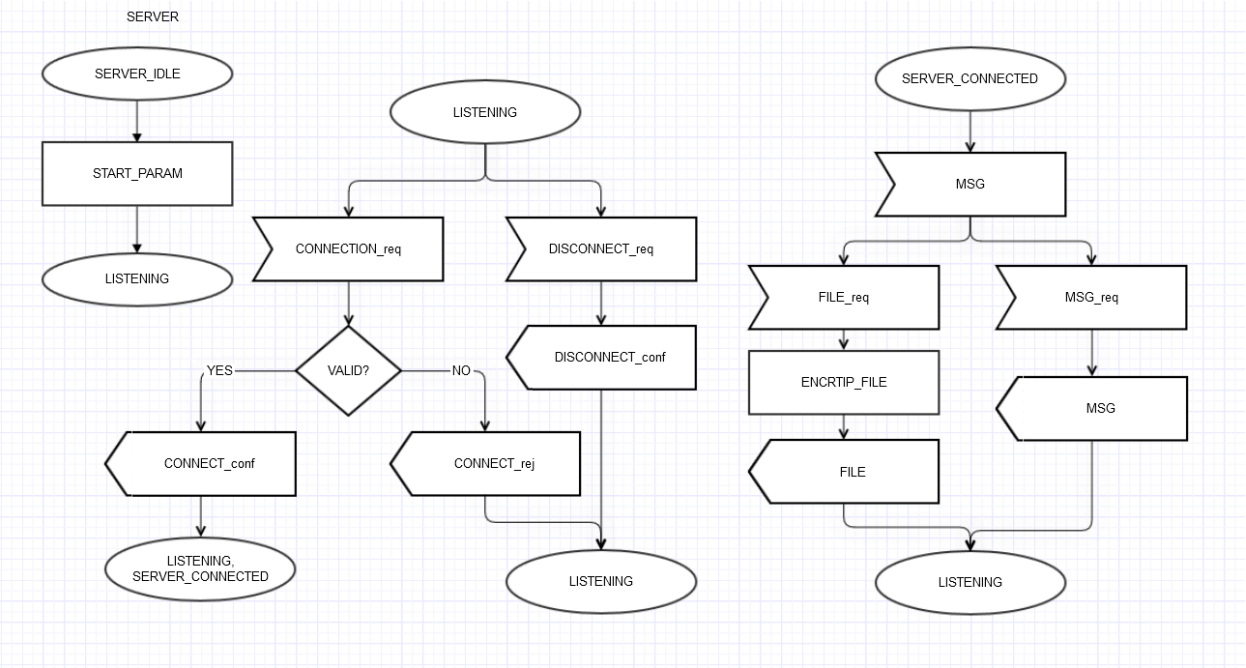
Slika1 – MSC dijagram

Konekcija se uspostavlja i zatvara tako što se između prijemne i predajne strane razmene poruke tri puta. U toku aktivne veze se vrši razmena poruka.

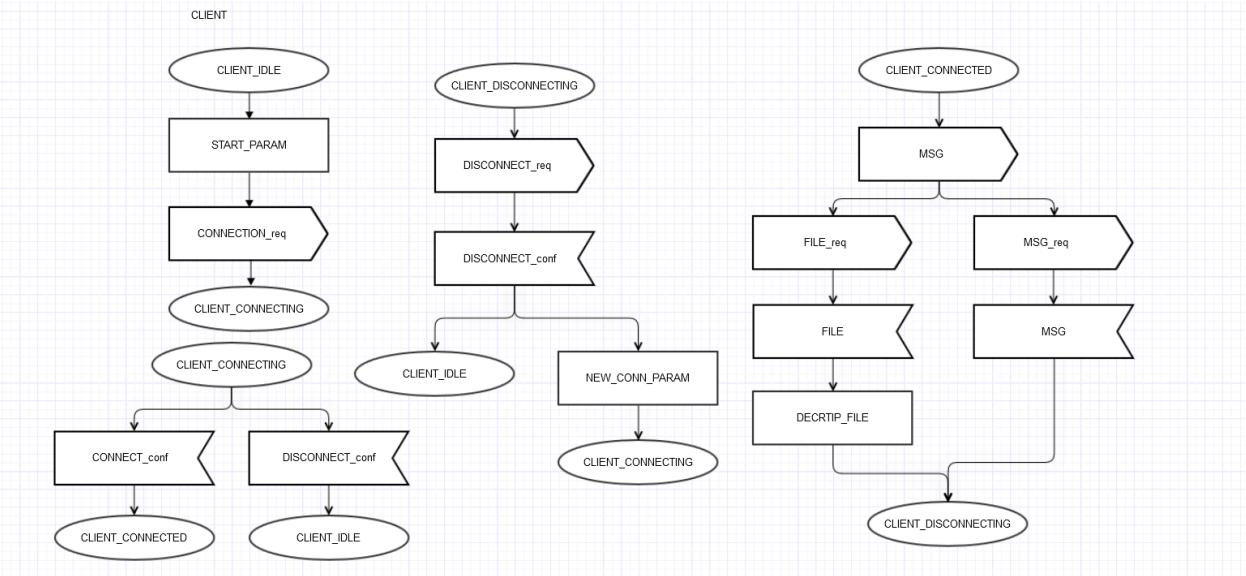
Trenutno je server napravljen tako što otvara dve utičnice. Nakon toga stoji u konstantom stanju slušanja, pomoću niti omogućen je priključivanja više klijenta. Klijent

nakon uspešo uspostavljenje veza trebalo bi da traži određeni broj portova. U ovom slučaju je realizovan da od servera dobija samo jedan. Uspostavi se nova veza preko kojeg se šalje datoteka.

Pre slanja server enkoduje tekstualni fajl Monoalfabetskom šifrom. Ključ šifre je 5, tj. a se preslikava na f, b na g itd. Ovo je realizovan iskorišćenjem ASCII tabele. Dekodovanje se vrši inverzno, nakon primanja datoteke.



Slika2 – SDL dijagrami servera



Slika1 – SDL dijagrami klijenta

4. Testiranje

Uz server je napravljen IPV4 i IPV6 klijent sa istom funkcionalnostima. Pomoću njih je potvrđen ispravnost rada više veza na različitih utičnica, enkripcije/dekripcije i dual stack funkcionalnost.

5. Zaključak

I ako formiranje više istovremenih veza za prenos datoteka u više delova nije implementirano, logično je da bi mnogo ubrzao proces slanja. Očekuje se da kod manjih datoteka ubrzanja je minimalna, kod većih značajnija. Nakon određenog broja veza očekuje se da se smanji značajnost ubrzanja, da stagnira. Moguće je i da smanji efikasnost.

6. Literatura

[1] *Materijali sa vežbi (Ipv6.pdf, Enkripcija.pdf)*, Fakultet Tehničkih Nauka, Računarska tehnika i računarske komunikacije, MRKiRM1, 2021-2022

[2] *Transmisioni kontrolni protokol*, Vikipedija Slobodna enciklopedija, https://sr.wikipedia.org/sr-el/transmisioni_kontrolni_protokol