Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií
Ilkovičova 2, 842 16 Bratislava 4
9. 12. 2021

# Komunikácia s využitím UDP protokolu

Filip Remšík

1. Zadanie

   Navrhnite a implementujte program s použitím vlastného protokolu nad protokolom UDP (User Datagram Protocol) transportnej vrstvy sieťového modelu TCP/IP. Program umožní komunikáciu dvoch účastníkov v lokálnej sieti Ethernet, teda prenos textových správ a ľubovoľného súboru medzi počítačmi (uzlami).
   Program bude pozostávať z dvoch častí – vysielacej a prijímacej. Vysielací uzol pošle súbor inému uzlu v sieti. Predpokladá sa, že v sieti dochádza k stratám dát. Ak je posielaný súbor väčší, ako používateľom definovaná max. veľkosť fragmentu, vysielajúca strana rozloží súbor na menšie časti - fragmenty, ktoré pošle samostatne. Maximálnu veľkosť fragmentu musí mať používateľ možnosť nastaviť takú, aby neboli znova fragmentované na linkovej vrstve.
   Ak je súbor poslaný ako postupnosť fragmentov, cieľový uzol vypíše správu o prijatí fragmentu s jeho poradím a či bol prenesený bez chýb. Po prijatí celého súboru na cieľovom uzle tento zobrazí správu o jeho prijatí a absolútnu cestu, kam bol prijatý súbor uložený.
   Program musí obsahovať kontrolu chýb pri komunikácii a znovuvyžiadanie chybných fragmentov, vrátane pozitívneho aj negatívneho potvrdenia. Po prenesení prvého súboru pri nečinnosti komunikátor automaticky odošle paket pre udržanie spojenia každých 5-20s pokiaľ používateľ neukončí spojenie. Odporúčame riešiť cez vlastne definované signalizačné správy.

2. Štruktúra hlavičky

| Info | Poradie fragmentu | Veľkosť fragmentu | Checksum | Dáta |
|---|---|---|---|---|
| 1b | 2b | 2b | 4b | 1463 a menej |

   Info-informácie o type paketu (0-naviazanie spojenia, 1-nasledovať bude prenos súboru, 2-nasledovať bude prenos správy, 3 prenos dát,4- správny prenos,5-chybný prenos,6-koniec prenosu,7-udržanie spojenia,8-prepnutie,9-koniec)
   Poradie fragmentu-poradové číslo práve posielaného fragmentu
   Veľkosť fragmentu-nastavená veľkosť fragmentu
   Checksum-údaj pre kontrolu správnosti posielanej časti
   Data -posielaná časť súboru, názov súboru
   Maximálna veľkosť dát (fragmentu): 1463 bajtov
   Zmeny oproti návrhu: veľkosť fragmentu bolo zmenená z 3b na 2b a priestor pre checksum bol zväčšený z 2 na 4 b, z dôvodu že výstupom z crc 32 je 32 bitové číslo.
3. Checksum
   Na kontrolovanie správneho doručenia dát použijem metódu CRC, konkrétne crc 32. Dáta sú spracovávané bit po bite, je na ne aplikovaný bitový posun a je medzi nimi vykonávaná operácia xor. Výstupom z funkcie je 32 bitové číslo. Checksum je vykonávaný na oboch uzloch, pričom prijímací uzol porovnáva výsledok s tým ktorý mu bol poslaný od vysielacieho uzlu.

4. ARQ
   Funkcia na kontrolu správnosti poslaného paketu.
   Stop and wait
   Metóda na posielanie informácií medzi odosielateľom a prijímateľom. Posielanie správ nieje taká rýchle ale pri náhodnej chybe je jej odstránenie jednoduchšie. Odosielateľ odošle v daný čas jeden dátový paket a čaká kým mu prijímateľ potvrdí správne prijatie paketu a až potom môže poslať ďalší paket. Ak dostane správu o poškodenom prenose pokúsi sa paket znovu poslať.

5. Udržanie spojenia
   V prípade že medzi uzlami nebude prebiehať posielanie žiadneho súboru bude odosielateľ v pravidelných časových intervaloch ( 10 s) posielať správu na overenie spojenia a čakať na odpoveď. V prípade že sa používateľ rozhodne pre nejakú z ďalších možností posielania alebo výmeny tak sa udržiavanie spojenia zastaví a spustí sa až keď sa zadaná úloha ukončí.

6. Simulácia chyby
   Simuláciu chyby si môžem nastaviť na každom x- tom poslanom pakete. Na úpravu správnych dát sa použije bitový posun doprava. Upravené dáta sú následne poslane na prijímací uzol a ten po overení zistí že sú zlé a následne požiada o znovuodoslanie paketu. Následne sa už pošle správny paket a počká sa na overenie aj jeho správnosti.

7. Požité knižnice
   Binascii-na použitie crc32 metódy
   Socket-celková komunikácia pomocou UDP protokolu
   Os- informácie o súbore
   Threading- bežanie viacerých procesov súčastne
   Time-na čakanie
8. Ukážka wiresharku-posielaná bola len hlavička s údajmi o type prenosu

udp.port==4596

| No. | Time | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|---|
| 1 | 0.000000 | 127.0.0.1 | 127.0.0.1 | UDP | 41 | 56065 → 4596 Len=9 |
| 2 | 0.000149 | 127.0.0.1 | 127.0.0.1 | UDP | 41 | 4596 → 56065 Len=9 |
| 3 | 10.003871 | 127.0.0.1 | 127.0.0.1 | UDP | 41 | 56065 → 4596 Len=9 |
| 4 | 10.004010 | 127.0.0.1 | 127.0.0.1 | UDP | 41 | 4596 → 56065 Len=9 |
| 10 | 20.019017 | 127.0.0.1 | 127.0.0.1 | UDP | 41 | 56065 → 4596 Len=9 |

9. Používateľské rozhranie
   Po zapnutí programu je možné si navoliť či chceme ďalej pokračovať ako vysielač alebo prímač. Po zvolení prímača si môžeme nastaviť IP a port. Po zvolení vysielača si môžeme nastaviť IP a port a následne si zvoliť či chceme posielať súbor alebo správu:
   Súbor-môžeme si nastaviť názov posielaného súboru a navoliť si veľkosť fragmentu

   Vzor výpisu po prijatí súboru

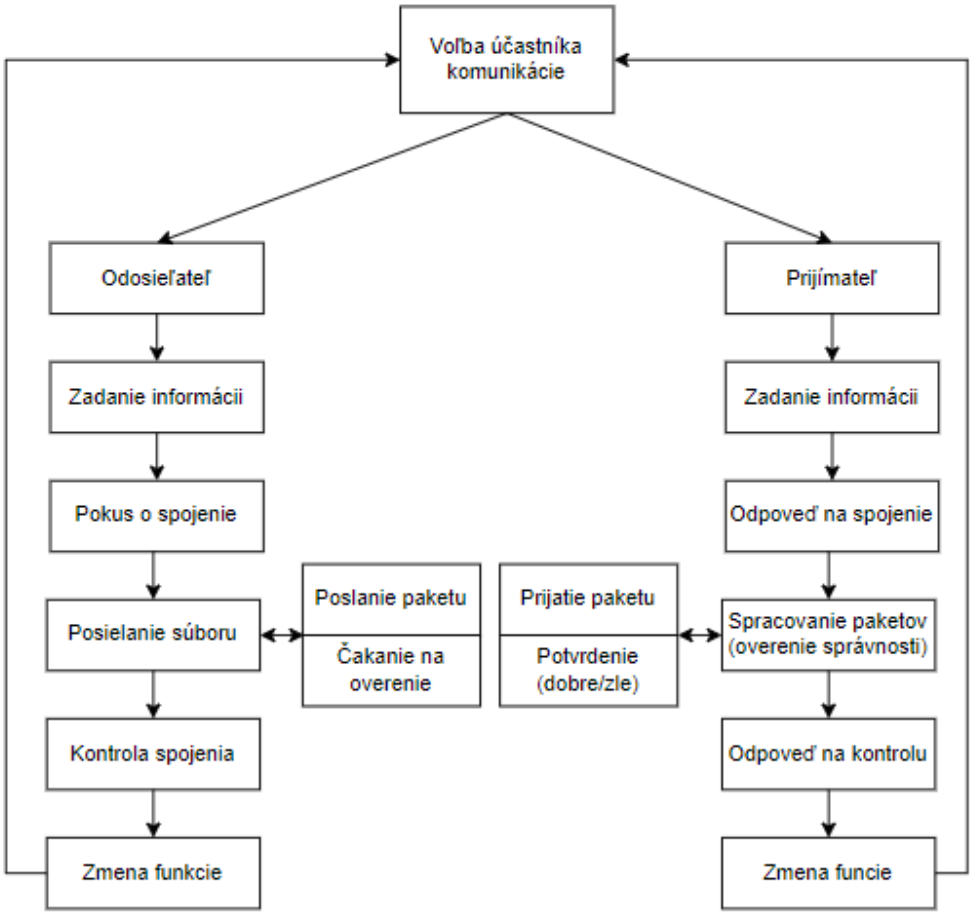
```
60  chyba
60  dobre
61  dobre
62  dobre
63  dobre
64  dobre
65  dobre
Cesta:  C:\Users\42190\PycharmProjects\Komunikátor Názov: copy_foto.jpg
Veľkosť súboru 64806  B
Dobre poslané pakety:  65
Zle poslané pakety:  6
```

Správa-môžeme si nastaviť veľkosť fragmentu a zadať správu.
Následne je už možné okrem ďalšieho posielania možnosť prepnúť na opačnú funkciu alebo komunikáciu medzi uzlami ukončiť.

10. Implementačné prostredie
    Program je implementovaný v prostredí pycharm a pracuje s jazykom python na verzii 3.9.7

11. Záver
    Program dokáže posielať dáta medzi uzlami ako meniť svoju funkciu. Pri testovaní som posielal rôzne typy súborov s maximálnou veľkosťou okolo 5 B a nebol s tým žiaden problém. Programu niekedy chvíľu trvá kým po načítaní vstupu vykoná danú operáciu, príčinou by malo byť dlhšie zastavovanie metódy na udržiavanie spojenia.

12. Diagram


Voľba účastníka komunikácie
Odosieľateľ
Prijímateľ
Zadanie informácii
Zadanie informácii
Pokus o spojenie
Odpoveď na spojenie
Posielanie súboru
Poslanie paketu
Čakanie na overenie
Prijatie paketu
Potvrdenie (dobre/zle)
Spracovanie paketov (overenie správnosti)
Kontrola spojenia
Odpoveď na kontrolu
Zmena funkcie
Zmena funcie