**Slovenská technická univerzita v Bratislave**

Fakulta informatiky a informačných technológií

Počítačové a komunikačné siete

2024/25

Zadanie – Komunikácia s využitím UDP protokolu

Obsah

[Protokol UDP 3](#_Toc180447013)

[Návrh programu 4](#_Toc180447014)

[Hlavička protokolu 5](#_Toc180447015)

[ARQ metóda 6](#_Toc180447016)

[Keep-alive metóda 7](#_Toc180447017)

[Flowchart 8](#_Toc180447018)

[Použité knižnice 9](#_Toc180447019)

# Protokol UDP

UDP protokol sa vyznačuje hlavne jeho rýchlosťou. Keďže protokol neoveruje inicializovanie alebo skončenie spojenia 3 / 4 way handshake-om, prenos dát je oveľa rýchlejší. Nevýhodou je, že spoľahlivý prenos dát nie je zaručený. Používa sa hlavne v prípadoch, kedy je potrebný rýchly prenos dát bez overovania a preposielania chybných dát (napr. streamovanie).

UDP na prenos dát používa tzv. datagramy. Maximálna veľkosť datagramu je 65 535B. Taktiež musíme počítať s tým, že sa v ňom nachádzajú hodnoty, ktoré používajú prenosové protokoly = UDP, IP, a v našom prípade aj vlastný custom header. Preto sa veľkosť zmení z 65 535 na -> 65535 – UDP Header: 8B – IP Header: 20B – Custom header: 8B = 65 535 – 8 – 20 – 8 = 65 499B voľných na prenos dát.

Linková vrstva má v sebe zahnutú fragmentáciu, a kedže nechceme, aby táto fragmentácia nastala, musíme dať maximálny limit packetu na 1500B. Preto, maximálna veľkosť môže byť 1500 – 8 – 20 – 8 = 1464B.

# Návrh programu

Používateľ zadá svoj port, na ktorom chce počúvať. Inicializuje spojenie, kde medzi dvoma bodmi komunikácie prebehne three-way handshake. Po potvrdení a nadviazaní spojenia je spustené Chatové GUI, kde užívatelia posielajú a prijímajú správy. Ďalej, používateľ vyberie, či chce odoslať správu alebo súbor. Po zadaní vstupu (správa/súbor) program overí, či je potrebné vstupné dáta fragmentovať. Pokiaľ áno, rozdelí správu do fragmentov, a nakoniec správu odošle. Na strane prijímateľa sa dáta preberú, overia sa pomocou CRC, a vyžiada si chybné fragmenty vybranou ARQ metódou, a dáta sa dopošlú. Po správnom doručení dát sa dáta zobrazia (súbory uložia). Používateľ môže posielať správy až dokým sa nerozhodne vymeniť strany komunikácie alebo komunikáciu prerušiť. Na to, aby sme držali komunikáciu aktívnu, sa bude používať keep-alive packet každých X sekúnd.

# Hlavička protokolu

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Typ | Poradie fragmentu | Informácia o ďalšom fragmente | Dĺžka dát | CRC | Dáta |
| 1B | 2B | 1B | 2B | 2B | XB |

**Typ:**

|  |  |  |
| --- | --- | --- |
| Hodnota decimálna | Hodnota binárna | Význam flagu |
| 1 | 00000001 | Inicializuj spojenie |
| 2 | 00000010 | Prenos správy |
| 3 | 00000011 | Prenos obrázku |
| 4 | 00000100 | Keep alive |
| 5 | 00000101 | CRC NOK |
| 6 | 00000110 | Acknowledge |
| 7 | 00000111 | ARQ |
| 8 | 00001000 | Ukonči spojenie |

**Poradie** **fragmentu** – značí o poradí fragmentu celej fragmentovanej správy. Využívané v ARQ.

**Informácia o ďalšom fragmente:**

|  |  |
| --- | --- |
| Hodnota | Význam |
| 01 | Očakávaj ďalší fragment |
| 10 | Neočakávaj ďalší fragment |

**CRC checksum** – Cyclic redundancy check je metóda na overovanie správnosti odoslaných/prijatých dát. Pre obe strany stanovíme deliteľ polynómu, napríklad x3 + x + 1. Z polynómu získame dĺžku K. Na strane odosielateľa pridáme k-1 núl na koniec dát. Následne dáta vydelíme pomocou polynómu a zvyšok po delení uložíme a pripojíme ho na koniec. Na strane príjemcu vykonáme delenie znova a ak dostaneme k-1 núl na konci, údaje boli odoslané bez chýb.

# ARQ metóda

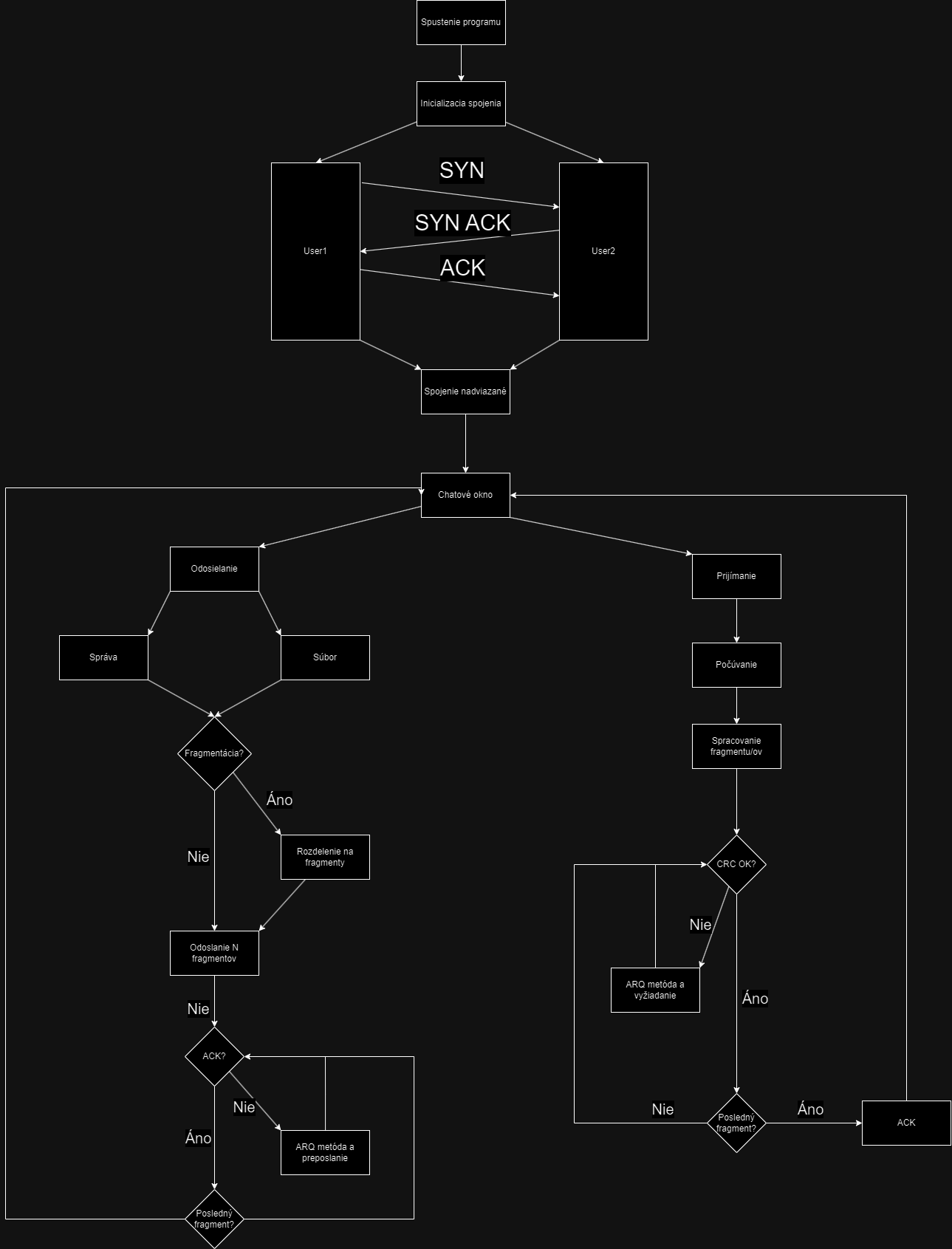
* Stop & wait
* Go Back-N
* **Selective Repeat**
* Vylepšená Go Back-N/Selective Repeat

Na začiatku je naplánované spraviť Selective Repeat ARQ metódu. Spočíva v tom, že odosielateľ posiela fragmenty. Prijímateľ si podľa poradia vyskladá správu, a zistí, či mu prišli všetky dáta správne. Ak nie, požiada o dané fragmenty ešte raz.

# Keep-alive metóda

Keep-alive sa používa na udržanie spojenia. Túto správu posiela strana odosielateľa a prijímateľ na tieto správy odpovedá. Týmto je zaručené overenie aktivity oboch strán. Pokiaľ sa keep-alive správa neprijme na strane prijímateľa do X sekúnd alebo odosielateľovi nepríde potvrdenie na keep-alive správu do X sekúnd, komunikácia sa ukončí.

# Flowchart



# Použité knižnice

Socket – komunikácia pomocou socketov

Theading – multithreading pre listening na správy, keep-alive

Sys – command line argumenty pre spustenie aplikácie v GUI

PyQt – Chat GUI

Colorama – zvýraznenie určitých výpisov do konzole