

# Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

Počítačové a komunikačné siete

Analýzátor sieťovej komunikácie

## Contents

1. Zadanie úlohy .....	4
2. Štruktúra súborov na prácu s portami a protokolmi .....	5
3. Práca s programom .....	5
4. Flowchart načítania programu a určenia typu .....	7
5. Flowchart pre Ethernet II atribúty .....	8
6. Komunikácie .....	9
7. Zvolené prostredie .....	10
8. Záver a možné zlepšenie programu .....	10



## 1. Zadanie úlohy

Návrh a implementácia analyzátora Ethernet siete na záznam komunikácie v súbore .pcap a poskytovanie nasledujúcich informácií o komunikácii. Výstup sa exportuje do formátu YAML.

### Funkcionalita:

- Zoznam všetkých rámcov v hexadecimálnom tvare postupne, ako boli zaznamenané v súbore.
- Zoznam IP adries a zapuzdreného protokolu na vrstvách 2-4 pre rámce Ethernet II.
- Poskytnite nasledujúcu štatistiku pre pakety IPv4 na konci výstupu z úlohy 2:
- Program s analýzou komunikácie pre vybrané protokoly.

Tento popis zabezpečuje, že analyzátor bude schopný spracovať záznamy komunikácie v súbore .pcap a poskytne informácie jednotlivých rámcov. Výstup bude vo formáte YAML, ktorý je ľahko spracovateľný pre ďalšiu analýzu alebo spracovanie dát.

### Výpis informácií rámca:

- Poradové číslo rámca
- Dĺžka rámca
- Typ rámca
- Sap (IEEE 802.3 LLC)
- Pid (IEEE 802.3 LLC + SNAP)
- Zdrojová MAC adresa
- Cieľová MAC adresa
- Ethertype
- Zdrojová IP adresa
- Cieľová IP adresa
- Protokol
- Zdrojový port
- Cieľový port
- Aplikačný protokol
- Dáta v hexadecimálnom tvare

Taktiež, okrem výpisu je potrebné implementovať:

- Načítanie dát pre prácu s protokolmi a portami z externých súborov
- Štatistiku IP adries
- Filter

## 2. Štruktúra súborov na prácu s portami a protokolmi

Z projektového adresára majú tieto súbory cestu /type\_files/'subor'.txt

Ether\_type.txt:

```
ether_type:
  0806: ARP
  0800: IPv4
  88CC: LLDP
  86DD: IPv6
  9000: ECTP
```

## 3. Práca s programom

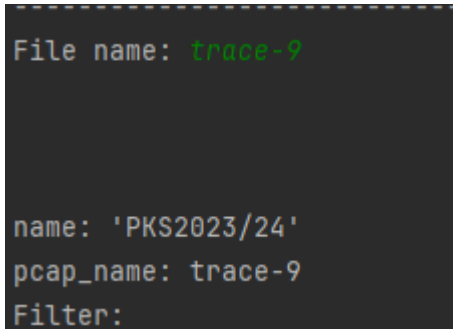
Po spustení programu sú používateľovi do konzoly vypísané všetky možné súbory .pcap, ktoré môže načítať.

```
C:\Users\Lukáš\AppData\Local\Microsoft\WindowsApps\python3.11.exe -m "Zadanie 1"
Choose file:
eth-1.pcap
eth-2.pcap
eth-3.pcap
eth-4.pcap
eth-5.pcap
eth-6.pcap
eth-7.pcap
eth-8.pcap
eth-9.pcap
trace-1.pcap
trace-10.pcap
trace-11.pcap
trace-12.pcap
trace-13.pcap
trace-14.pcap
trace-15.pcap
trace-16.pcap
trace-17.pcap
trace-18.pcap
trace-19.pcap
trace-2.pcap
trace-20.pcap
trace-21.pcap
trace-22.pcap
trace-23.pcap
trace-24.pcap
trace-25.pcap
trace-26.pcap
trace-27.pcap
trace-3.pcap
trace-4.pcap
trace-5.pcap
trace-6.pcap
trace-7.pcap
trace-8.pcap
trace-9.pcap
trace_ip_nad_20_B.pcap
```

Lukáš Lovás

ID: 120964

Následne si program vypýta od používateľa vstup, a to je meno súboru bez koncovky .pcap.

A screenshot of a terminal window with a dark background. The text is displayed in a monospaced font. The first line is 'File name: trace-9' where 'trace-9' is in green. The second line is 'name: 'PKS2023/24''. The third line is 'pcap\_name: trace-9'. The fourth line is 'Filter:'.

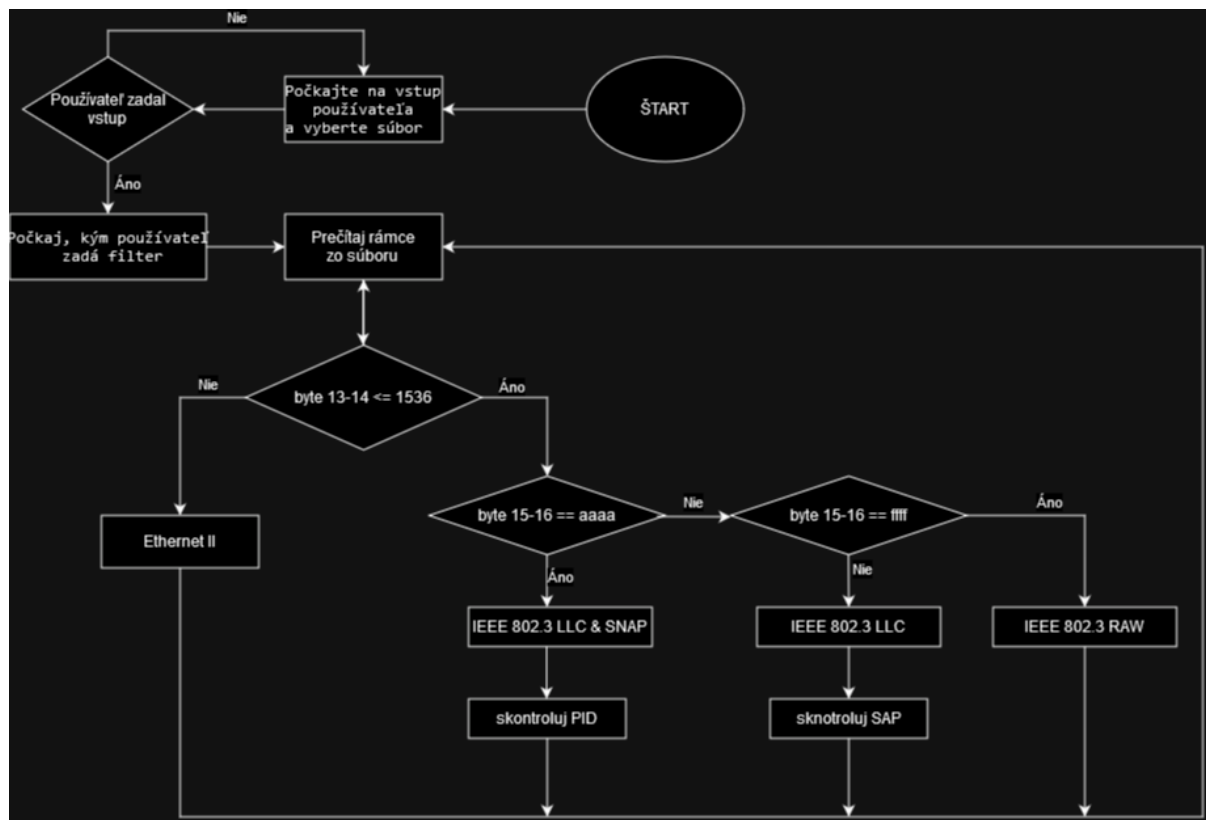
```
File name: trace-9
name: 'PKS2023/24'
pcap_name: trace-9
Filter:
```

Ako ďalšie si program od vstup pre filter.

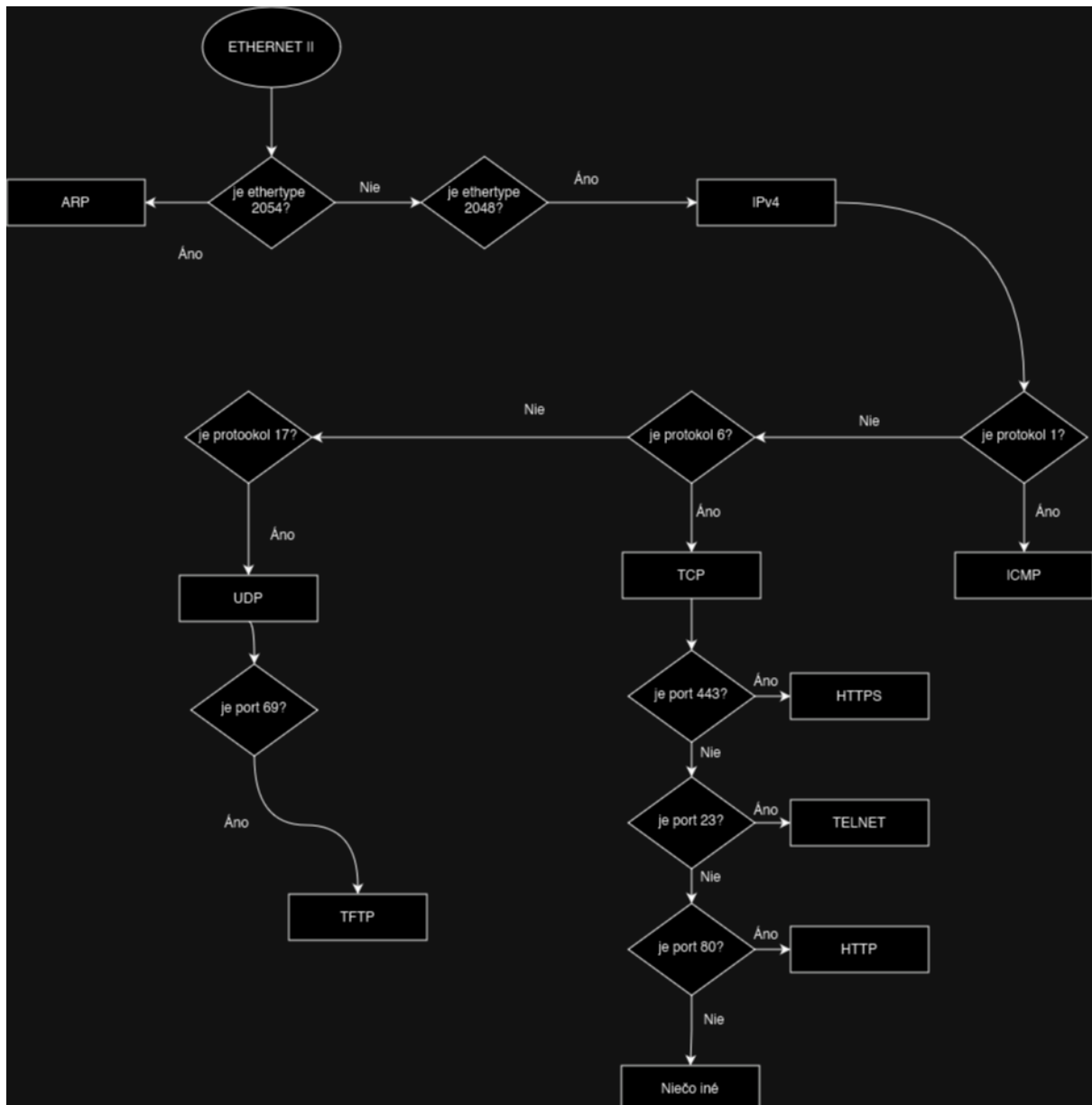
Pri stlačení klávesy ENTER keď si program vypýta vstup pre filter, sa filter uloží ako prázdny string, čo indikuje klasický výpis (Zadanie 1) pre určený .pcap súbor

Následne už program robí analýzu rámcov podľa zadaného filtra a výpis do .yaml súboru, ktorý sa vytvára do projektového adresára s názvom output.txt.

#### 4. Flowchart načítania programu a určenia typu



## 5. Flowchart pre Ethernet II atribúty





## 6. Komunikácie

### - Komunikácia TCP protokolov (Connection-oriented protocol communication):

Ako prvé sa z vyfiltrovaných rámcov urobia páry portov (tuple), ktoré sa uložia do listu. Následne sa pre každú komunikáciu overia „handshake-y“. 3-way handshake otvára komunikáciu, a preto sa overuje ako prvý. Začne for each cyklus rámcov, kde sa overuje, či sú z streamu momentálnej komunikácie, a ak áno, pozerá sa 48. bajt v hexadecimálnom kóde rámca. Následne sa podľa stavových flagov overujú postupne bajty 2, 18, 16 = SYN, SYN/ACK, ACK.

Pokiaľ sa úspešne našli tieto 3 bajty, nastavíme flag pre 3-way handshake na true a ideme overovať 4-way closing handshake.

Pre 4-way handshake som zvolil inú stratégiu na overovanie, kvôli množstvu kombinácií, ktorými sa môže komunikácia ukončiť.

Najprv sa for each cyklom prejdú všetky rámce, kde overujeme, či sú z streamu komunikácie. Následne sa pre každý rámec odstrihne z hexadecimálneho kódu 48. bajt, ktorý sa prilepí k pomocnému stringu. Nasledovný string sa rozdelí, a získavame pole bajtov, ktoré taktiež uložíme odzadu, aby sa ľahšie pristupovalo k hodnotám, ktoré sú z konečných rámcov. Kombinácie ktoré sú v if podmienkach obsiahnuté, sú:

- RST
- RST/ACK
- FIN/ACK,FIN/ACK
- FIN/ACK/PUSH,FIN/ACK/PUSH,ACK
- FIN/ACK,ACK,FIN/ACK,ACK
- FIN,FIN/ACK,ACK

Ak sa potvrdí ukončenie komunikácie, flag pre 4-way handshake sa nastaví na True a metóda vracia nasledovne:

```
if three_way_handshake_flag and four_way_handshake_flag:
    return True
if three_way_handshake_flag and not four_way_handshake_flag:
    return 0
else:
    return False
```

-Vráti True, pokiaľ je komunikácia kompletná,

-Vráti 0, pokiaľ je nekompletná,

-Vráti False

Pokiaľ sa z metódy na overenie handshake-ov vráti True, vyhľadajú sa všetky rámce, ktoré boli pre túto komunikáciu použité, a vytvorí sa objekt Communication, ktorý reprezentuje úplnú komunikáciu a obsahuje všetky potrebné atribúty pre exportovanie do YAML. Ak sa vráti 0, vytvorí sa objekt Communication, ktorý reprezentuje neúplnú komunikáciu a obsahuje všetky potrebné atribúty pre exportovanie do YAML.

Lukáš Lovás

ID: 120964

- Komunikácia UDP protokolov:

Vynechané

- Komunikácia ICMP protokolov:

Vynechané

- Komunikácia ARP protokolov:

Vynechané

- IP fragmentácia:

Vynechané

## 7. Zvolené prostredie

Implementácia bola napísaná v Pythone v prostredí The JetBrains IDE Pycharm.

Ruamel yaml bol použitý na export YAML.

## 8. Záver a možné zlepšenie programu

- Doimplementovanie ostatných komunikácií
- Vytvorenie GUI, alebo zlepšenie konzolového menu
- Support pre viaceré filtre pre jeden súbor