Fakulta informatiky a informačných technológií STU v Bratislave

Ilkovičova 2, 842 16 Bratislava 4

Počítačové a komunikačné siete

Analyzátor sieťovej komunikácie

Ondrej Špánik  
ID: 103151

Meno cvičiaceho: Ing. Lukáš Mastiľak

Časy cvičení: Štvrtok 18:00

Akademický rok: 2022/23 ZS

Obsah

[1. Koncepcia riešenia 3](#_Toc117115672)

[2. Mechanizmus analyzovania protokolov 4](#_Toc117115673)

[Vrstva sieťového rozhrania (ETH) 4](#_Toc117115674)

[Sieťová vrstva (IP, ICMP) 4](#_Toc117115675)

[Transportná vrstva (TCP, UDP) 5](#_Toc117115676)

[Zoskupovanie podľa komunikácií 5](#_Toc117115677)

[3. Štruktúra externých súborov pre protokoly/porty 6](#_Toc117115678)

[4. Používateľské rozhranie 7](#_Toc117115679)

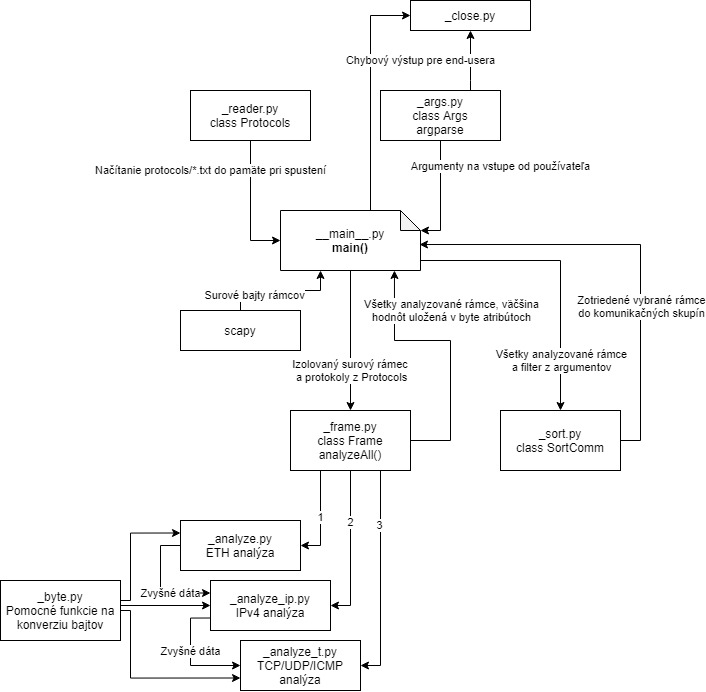
[Ukážkový štandardný výpis 7](#_Toc117115680)

[5. Implementačné prostredie 8](#_Toc117115681)

[Prvé spustenie 8](#_Toc117115682)

[6. Zhodnotenie a možnosti rozšírenia 8](#_Toc117115683)

# Koncepcia riešenia

Blokový návrh fungovania

# Mechanizmus analyzovania protokolov

Podľa jednotlivých vrstiev:

## Vrstva sieťového rozhrania (ETH)

|  |  |
| --- | --- |
|  | Dôjde k rozpoznaniu druhu Ethernetového spojenia na základe štyroch podporovaných štandardov:   * 802.3 RAW * 802.3 LLC+SNAP * 802.3 LLC * ETHERNET 2   V niekorých daných štandardoch sú definované najpodstatnejšie zložky pre ďalšie pokračovanie analýzy:   * EtherType: nachádza sa na rôznych miestach ETH2 a LLC+SNAP, v ostatných nie je prítomný. Je potrebný pre identifikáciu IPv4 * Data: Zvyšné dáta, ktoré sa posúvajú do analýzy na ďalšej vrstve   V tomto kroku okrem iného sú uložené source a destination MAC adresy a v prípade RAW identifikovaný IPX. |

## Sieťová vrstva (IP, ICMP)

|  |  |
| --- | --- |
|  | Keďže na základe EtherType je známe IPv4, stačí vytiahnúť potrebné dáta:   * Protokol TCP/UDP/ICMP * Source IP adresa * Destination IP adresa   A pokračovať ďalej so zvyšnými dátami na TCP/UDP analýzu.  ICMP je riešený spoločne s TCP/UDP vzhľadom na jednoduchosť riešenia. |

## Transportná vrstva (TCP, UDP)

|  |  |
| --- | --- |
|  | Na základe protokolu obdržaného z hlavičky IPv4 ďalej pokračuje analýza.  V prípade TCP sú identifikované aj jednotlivé flagy, ktoré sú pomocou slovníka aj neskôr ľudsky popísané.  TCP flagy zdieľajú polovicu prvého bajtu s dĺžkou TCP hlavičky, ktorá na rozdiel od UDP hlavičky môže mať variabilnú dĺžku, minimálne avšak 20B. V kóde je možné vidieť bitové operácie pre vytiahnutie podstatných častí v oboch prípadoch.  Vďaka dĺžke TCP hlavičky je možné ďalej pokračovať v analýze dát, tam už na rad prichádza aplikačná vrstva s protokolmi ako HTTP, DHCP, FTP, atď. |

## Zoskupovanie podľa komunikácií

* Špecificky implementované a testované pre:
  + TFTP
  + ICMP
  + ARP
* Niektoré ďalšie protokoly tiež podporované
* Cieľom zoskupovania je informovať používateľa o všetkej komunikácií medzi istým klientom a serverom na danom protokole. Pre zoskupovanie treba použiť špeciálny argument pri spúšťaní programu.

# Štruktúra externých súborov pre protokoly/porty

Externé súbory je všetky možné nájsť v zložke protocols.

Pre správne fungovanie programu je potrebné aby všetky existovali pri spustení. Môžu byť aj prázdne.

V prípade chýbajúceho páru v danom súbore je predvolené informovať používateľa, že sa jedná o neznámy EtherType, TCP port a pod.

Súbory sú načítané hneď po spustení programu a následne sa k nim už nepristupuje v ciele zrýchlenia programu ako aj zamedzenia možnosti akéhokoľvek pádu z dôvodu spamového prístupu.

|  |  |
| --- | --- |
| eth\_types.txt | 0x???? Meno   * 0x???? je hexadecimálne znázornený EtherType * Meno je ľubovoľné pomenovanie pre daný EtherType * Zobrazené pre koncového používateľa v bežnom výpise analýzy z programu |
| icmp\_types.txt | n Popis   * n je dekadické číslo daného ICMP typu od ktorého sa odvodí popis * Popis je ľubovoľný zrozumiteľný popis pre koncového používateľa |
| ip\_protocols.txt | 0x?? Meno   * 0x?? je hexadecimálne znázornený protokol pod IP * Najpoužívanejšie a ďalej analyzované sú ICMP, TCP, UDP, ktoré sú prvotne identifikované na základe 0x?? čísla * Meno je používané pre koncového používateľa a v prípade ICMP aj pri argumente na zoskupovanie |
| saps.txt | 0x?? Meno   * 0x???? je hexadecimálne znázornený SAP * Meno je ľubovoľné pomenovanie pre daný SAP * Zobrazené pre koncového používateľa v bežnom výpise analýzy z programu |
| tcp\_ports.txt | n Skratka   * n je číslo portu * Skratka je jednoslovný názov, ktorý možno ľahko vložiť do argumentu pre zoskupovanie (-s Skratka), na základe ktorého sa spätne určí „n“, teda číslo portu z tohto súboru a vykoná zoskupovanie |
| udp\_ports.txt | Rovnako ako pri tcp\_ports.txt |

# Používateľské rozhranie

Jedná sa o CLI rozhranie, ku ktorému je prístup cez argumenty programu.

Najpodstatnejší argument je cesta k PCAP súboru, ktorá má byť uvedená ako prvá.

Zvyšné argumenty a popis k nim je možné pozrieť pomocou flagu „-h“, ktorý otvorí vždy pomocníka:

Text

Description automatically generated

## Ukážkový štandardný výpis

|  |  |
| --- | --- |
| name: PKS2022/23  pcap\_name: \_Samples\trace-26.pcap  packets:  - frame\_number: 1  len\_frame\_pcap: 352  len\_frame\_medium: 356  frame\_type: Ethernet II  src\_mac: 00:16:47:02:24:1a  dst\_mac: 01:80:c2:00:00:0e  ether\_type: LLDP  hexa\_frame: |  01 80 c2 00 00 0e 00 16 47 02 24 1a 88 cc 02 07  04 00 16 47 02 24 00 04 07 05 46 61 30 2f 32 34  06 02 00 78 0a 06 53 77 69 74 63 68 0c f7 43 69  73 63 6f 20 49 4f 53 20 53 6f 66 74 77 61 72 65  2c 20 43 33 35 36 30 20 53 6f 66 74 77 61 72 65  20 28 43 33 35 36 30 2d 49 50 53 45 52 56 49 43  45 53 4b 39 2d 4d 29 2c 20 56 65 72 73 69 6f 6e  20 31 32 2e 32 28 35 35 29 53 45 37 2c 20 52 45  4c 45 41 53 45 20 53 4f 46 54 57 41 52 45 20 28  66 63 31 29 0a 54 65 63 68 6e 69 63 61 6c 20 53  75 70 70 6f 72 74 3a 20 68 74 74 70 3a 2f 2f 77  77 77 2e 63 69 73 63 6f 2e 63 6f 6d 2f 74 65 63  68 73 75 70 70 6f 72 74 0a 43 6f 70 79 72 69 67  68 74 20 28 63 29 20 31 39 38 36 2d 32 30 31 33  20 62 79 20 43 69 73 63 6f 20 53 79 73 74 65 6d  73 2c 20 49 6e 63 2e 0a 43 6f 6d 70 69 6c 65 64  20 4d 6f 6e 20 32 38 2d 4a 61 6e 2d 31 33 20 31  30 3a 31 30 20 62 79 20 70 72 6f 64 5f 72 65 6c  5f 74 65 61 6d 08 10 46 61 73 74 45 74 68 65 72  6e 65 74 30 2f 32 34 0e 04 00 14 00 04 10 0c 05  01 0a 14 1e fe 03 00 00 00 01 00 fe 06 00 80 c2  01 00 01 fe 09 00 12 0f 01 03 6c 00 00 10 00 00  - frame\_number: 2  len\_frame\_pcap: 60  len\_frame\_medium: 64  frame\_type: IEEE 802.3 LLC  src\_mac: 00:16:47:02:24:1a  dst\_mac: 01:80:c2:00:00:00  sap: STP  hexa\_frame: |  01 80 c2 00 00 00 00 16 47 02 24 1a 00 26 42 42  03 00 00 00 00 00 80 01 00 16 47 02 24 00 00 00  00 00 80 01 00 16 47 02 24 00 80 1a 00 00 14 00  02 00 0f 00 00 00 00 00 00 00 00 00  - frame\_number: 3  len\_frame\_pcap: 60  len\_frame\_medium: 64  frame\_type: Ethernet II  src\_mac: 00:16:47:02:24:1a  dst\_mac: 00:16:47:02:24:1a  ether\_type: ECTP  hexa\_frame: |  00 16 47 02 24 1a 00 16 47 02 24 1a 90 00 00 00  01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 00 00 00 00  ...  ipv4\_senders:  - node: 10.20.30.1  number\_of\_sent\_packets: 91  - node: 10.20.30.254  number\_of\_sent\_packets: 20  - node: 0.0.0.0  number\_of\_sent\_packets: 3  max\_send\_packets\_by:  - 10.20.30.1 | Naľavo je štandardný výstup programu do YAML súboru.  ...  Na spodnej časti súboru je možné vidieť leaderboard odosielateľov. |

# Implementačné prostredie

Implementácia bola vykonaná v jazyku Python 3.10 pomocou VSCode na operačnom systéme Windows 11.

Medzi použité externé knižnice patrí scapy, ktoré je použité čiste za účelom získania surových bajtov jednotlivých rámcov pre vlastnú analýzu.

Zvyšok programu je programovaný pomocou Python tried, teda objektovo-orientovane.

## Prvé spustenie

Pre spustenie je ideálne sprevádzkovať virtuálny environment (venv) pomocou návodu priloženého v README.md, následne venv aktivovať a nainštalovať scapy[complete] pomocou pip.

# Zhodnotenie a možnosti rozšírenia

V momentálnom stave program disponuje dostatočne robustným riešením pre analýzu jednotlivých častí väčšiny rámcov, avšak pri rozširovaní za to platím najmä časom.

Prvoradý by mal byť refaktoring: Presun nového TFTP kódu pod Filters. Následne úprava spôsobu, akým Filters funguje, keďže mi príde zbytočne abstraktný a je časovo namáhavé s ním manipulovať.

Program sa dá ďalej rozšíriť čo sa týka nielen výpisov, ale aj jednotlivých filterov: Chýba napríklad TCP filter.