Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

Analyzátor sieťovej komunikácie

Zadanie 1

Predmet: Počítačové a komunikačné siete

Prednášajúci: prof. Ing. Ivan Kotuliak, PhD.

Cvičiaci: Ing. Matej Janeba

Cvičenie: piatok 10:00

Rok: 2021/22 **Meno a Priezvisko:** Tamás Szakál
ID: 97059

Obsah

1. Zadanie úlohy

Navrhnite a implementujte programový analyzátor Ethernet siete, ktorý analyzuje komunikácie v

sieti zaznamenané v .pcap súbore a poskytuje nasledujúce informácie o komunikáciách.

Vypracované zadanie musí spĺňať nasledujúce body:

1) Výpis všetkých rámcov v hexadecimálnom tvare postupne tak, ako boli zaznamenané v súbore.

Pre každý rámec uveďte:

a) Poradové číslo rámca v analyzovanom súbore.

b) Dĺžku rámca v bajtoch poskytnutú pcap API, ako aj dĺžku tohto rámca prenášaného po

médiu.

c) Typ rámca – Ethernet II, IEEE 802.3 (IEEE 802.3 s LLC, IEEE 802.3 s LLC a SNAP, IEEE 802.3

– Raw).

d) Zdrojovú a cieľovú fyzickú (MAC) adresu uzlov, medzi ktorými je rámec prenášaný.

Vo výpise jednotlivé bajty rámca usporiadajte po 16 alebo 32 v jednom riadku. Pre prehľadnosť

výpisu je vhodné použiť neproporcionálny (monospace) font.

2) Pre rámce typu Ethernet II a IEEE 802.3 vypíšte vnorený protokol. Študent musí vedieť vysvetliť,

aké informácie sú uvedené v jednotlivých rámcoch Ethernet II, t.j. vnáranie protokolov ako aj

ozrejmiť dĺžky týchto rámcov.

3) Analýzu cez vrstvy vykonajte pre rámce Ethernet II a protokoly rodiny TCP/IPv4:

Na konci výpisu z bodu 1) uveďte pre IPv4 pakety:

a) Zoznam IP adries všetkých odosielajúcich uzlov,

b) IP adresu uzla, ktorý sumárne odoslal (bez ohľadu na prijímateľa) najväčší počet paketov

a koľko paketov odoslal (berte do úvahy iba IPv4 pakety).

IP adresy a počet odoslaných / prijatých paketov sa musia zhodovať s IP adresami vo výpise

Wireshark -> Statistics -> IPv4 Statistics -> Source and Destination Addresses.

4) V danom súbore analyzujte komunikácie pre zadané protokoly:

a) HTTP

b) HTTPS

c) TELNET

d) SSH

e) FTP riadiacef) FTP dátové

g) TFTP, uveďte všetky rámce komunikácie, nielen prvý rámec na UDP port 69

h) ICMP, uveďte aj typ ICMP správy (pole Type v hlavičke ICMP), napr. Echo request, Echo

reply, Time exceeded, a pod.

i) Všetky ARP dvojice (request – reply), uveďte aj IP adresu, ku ktorej sa hľadá MAC (fyzická)

adresa a pri ARP-Reply uveďte konkrétny pár - IP adresa a nájdená MAC adresa. V prípade,

že bolo poslaných viacero rámcov ARP-Request na rovnakú IP adresu, vypíšte všetky. Ak

sú v súbore rámce ARP-Request bez korešpondujúceho ARP-Reply (alebo naopak ARPReply bez ARP-Request), vypíšte ich samostatne.

Vo všetkých výpisoch treba uviesť aj IP adresy a pri transportných protokoloch TCP a UDP aj

porty komunikujúcich uzlov.

V prípadoch komunikácií so spojením vypíšte iba jednu kompletnú komunikáciu - obsahuje

otvorenie (SYN) a ukončenie (FIN na oboch stranách alebo ukončenie FIN a RST alebo ukončenie

iba s RST) spojenia a aj prvú nekompletnú komunikáciu, ktorá obsahuje iba otvorenie spojenia.

Pri výpisoch vyznačte, ktorá komunikácia je kompletná.

Ak počet rámcov komunikácie niektorého z protokolov z bodu 4 je väčší ako 20, vypíšte iba 10

prvých a 10 posledných rámcov tejto komunikácie. (Pozor: toto sa nevzťahuje na bod 1, program

musí byť schopný vypísať všetky rámce zo súboru podľa bodu 1.) Pri všetkých výpisoch musí byť

poradové číslo rámca zhodné s číslom rámca v analyzovanom súbore.

5) Program musí byť organizovaný tak, aby čísla protokolov v rámci Ethernet II (pole Ethertype),

IEEE 802.3 (polia DSAP a SSAP), v IP pakete (pole Protocol), ako aj čísla portov v transportných

protokoloch boli programom načítané z jedného alebo viacerých externých textových súborov.

Pre známe protokoly a porty (minimálne protokoly v bodoch 1) a 4) budú uvedené aj ich názvy.

Program bude schopný uviesť k rámcu názov vnoreného protokolu po doplnení názvu k číslu

protokolu, resp. portu do externého súboru. Za externý súbor sa nepovažuje súbor knižnice, ktorá

je vložená do programu.

6) V procese analýzy rámcov pri identifikovaní jednotlivých polí rámca ako aj polí hlavičiek

vnorených protokolov nie je povolené použiť funkcie poskytované použitým programovacím

jazykom alebo knižnicou. Celý rámec je potrebné spracovať postupne po bajtoch.

7) Program musí byť organizovaný tak, aby bolo možné jednoducho rozširovať jeho funkčnosť

výpisu rámcov pri doimplementovaní jednoduchej funkčnosti na cvičení.

8) Študent musí byť schopný preložiť a spustiť program v miestnosti, v ktorej má cvičenia. V prípade

dištančnej výučby musí byť študent schopný prezentovať podľa pokynov cvičiaceho program

online, napr. cez Webex, Meet, etc.

V danom týždni, podľa harmonogramu cvičení, musí študent priamo na cvičení doimplementovať

do funkčného programu (podľa vyššie uvedených požiadaviek) ďalšiu prídavnú funkčnosť.

Program musí mať nasledovné vlastnosti (minimálne):1) Program musí byť implementovaný v jazykoch C/C++ alebo Python s využitím knižnice

pcap, skompilovateľný a spustiteľný v učebniach. Na otvorenie pcap súborov použite

knižnice libpcap pre linux/BSD a winpcap/ npcap pre Windows. Použité knižnice a funkcie

musia byť schválené cvičiacim. V programe môžu byť použité údaje o dĺžke rámca zo struct

pcap_pkthdr a funkcie na prácu s pcap súborom a načítanie rámcov:

pcap_createsrcstr()

pcap_open()

pcap_open_offline()

pcap_close()

pcap_next_ex()

pcap_loop()

Použitie funkcionality libpcap na priamy výpis konkrétnych polí rámca (napr. ih->saddr) bude

mať za následok nulové hodnotenie celého zadania.

2) Program musí pracovať s dátami optimálne (napr. neukladať MAC adresy do 6x int).

3) Poradové číslo rámca vo výpise programu musí byť zhodné s číslom rámca v analyzovanom

súbore.

4) Pri finálnom odovzdaní, pre každý rámec vo všetkých výpisoch uviesť použitý protokol na 2. -

4. vrstve OSI modelu. (ak existuje)

5) Pri finálnom odovzdaní, pre každý rámec vo všetkých výpisoch uviesť zdrojovú a cieľovú

adresu / port na 2. - 4. vrstve OSI modelu. (ak existuje)

Nesplnenie ktoréhokoľvek bodu minimálnych požiadaviek znamená neakceptovanie riešenia

cvičiacim.

Súčasťou riešenia je aj dokumentácia, ktorá musí obsahovať najmä:

a) zadanie úlohy,

b) blokový návrh (koncepcia) fungovania riešenia,

c) navrhnutý mechanizmus analyzovania protokolov na jednotlivých vrstvách,

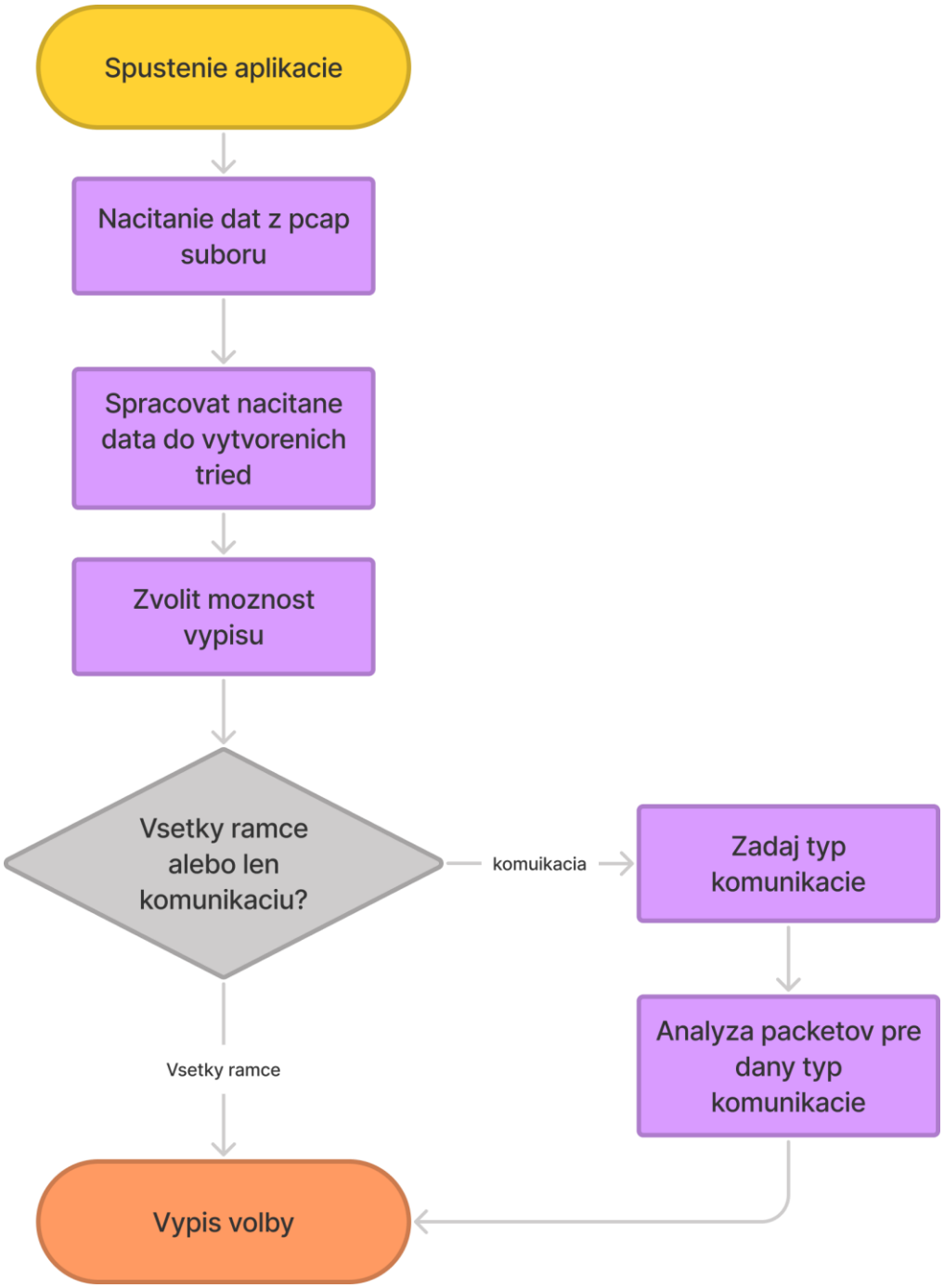
d) príklad štruktúry externých súborov pre určenie protokolov a portov,

e) opísané používateľské rozhranie,

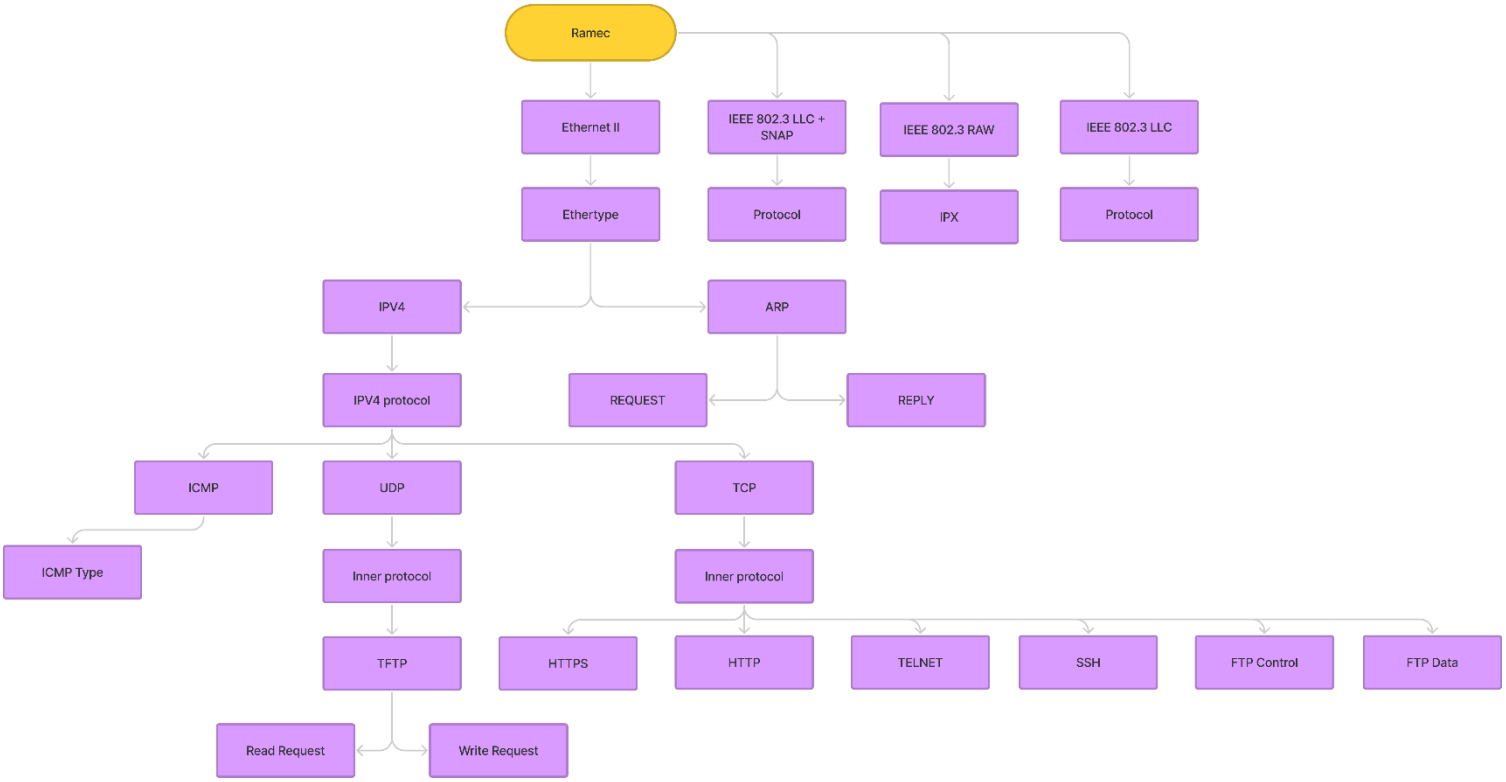
f) voľbu implementačného prostredia

2. Riešenie

2.1. Blokový návrh fungovania riešenia



2.2. Návrh mechanizmu analyzovania protokolov



2.3. Príklad štruktúry externých súborov

ether_types.txt - Notepad

File Edit Format View Help

```
0x200 XEROX PUP
0x201 PUP Addr Trans
0x800 Internet IP (IPv4)
0x801 X.75 Internet
0x805 X.25 Level 3
0x806 ARP (Address Resolution Protocol)
0x8035 Reverse ARP
0x809B Appletalk
0x80F3 AppleTalk AARP (Kinetics)
0x8100 IEEE 802.1Q VLAN-tagged frames
0x8137 NovellIPX
0x86DD IPv6
0x880B PPP
0x8847 MPLS
0x8848 MPLS with upstream-assigned label
0x8863 PPPoE Discovery Stage
0x8864 PPPoE Session Stage
0x9000 Loopback
```

2.4. Používateľské rozhranie

V prípade nášho Analyzátora sieťových komunikácii sa jedná o konzolovú aplikáciu. Naša konzolová aplikácia sa pýta otázky užívateľa na ktoré užívateľ buď odpovie sám(zadanie mena suborov) alebo aplikácia navrhne možnosti užívateľovi z ktorých si môže vybrať.

Naša aplikácia ako výstup analýzy používa výstupný súbor textový súbor ktorú si aplikácia vytvorí po prípade že daný výstupný súbor existuje, aplikácia ho prepíše.

Na začiatok sa naša aplikácia analyzovania sieťových komunikácii spýta používateľa z ktorého pcap súboru chce čítať dáta. Tieto pcap súbory musia byť umiestnení v priečinku pcap, inak aplikácia nenájde pcap súbor zadaný užívateľom. V prípade že nenájde daný pcap súbor aplikácia vypíše do konzoly „Súbor /názov zadaného súboru/ neexistuje“.

Používateľ ma v tomto prípade nekonečne veľa možnosti zadať názov súboru na analýzu. V prípade že používateľ chce skončiť a nechce analyzovať žiadny súbor stačí do konzoly zadať pri otázke „Zadaj meno súboru ktorý chceš analyzovať: “ ako názov súboru „exit“ a aplikácia skončí.

Ďalej sa aplikácia spýta užívateľa ako chce pomenovať výstupný súbor. V aplikácii pri zadávaní názvu výstupného súboru netreba pridať koncovku .txt aplikácia si to priloží sám pre komfort užívateľa.

Tento komfort však pre zadávanie mena súboru na analýzu neplatí. Používateľ musí v tomto prípade pridať koncovku k názvu súboru .pcap.

Ďalšia otázka konzolovej aplikácie je výberová. Používateľ si musí vybrať z možností na výpis. Nasledovne aplikácia vypíše zvolený výstup do výstupného súboru s názvom ktorý zadal používateľ sám. Po výpise do súboru aplikácia skončí. Medzi možnosťami je možnosť na skončenie aplikácie bez výpisu.

2.5. Implementačné prostredie

Aplikácia je implementovaná v programovacom jazyku Python použitím knižnice scapy, binascii, os.path v PyCharm 2021.2.2 (Professional Edition) s Python 3.9.

Diagramy k dokumentácii boli navrhnuté v aplikácii Figma.