Slovenská technická univerzita v Bratislave
Fakulta Informatiky a Informačných Technológií

Počítačové a komunikačné siete

Analyzátor sieťovej komunikácie

Emma Macháčová

Meno cvičiaceho : Ing. Lukáš Mastiľak
Čas cvičení : Štvrtok 16:00
Dátum vytvorenia : 17. Okt. 2021

Obsah

Cieľ práce

Cieľom je navrhnutie a implementácia programového analyzátora Ethernet siete, ktorý analyzuje komunikácie v sieti zaznamenané v .pcap súbore a poskytuje nasledujúce informácie o týchto komunikáciách:

- Výpis všetkých rámcov v hexadecimálnom tvare postupne tak, ako boli zaznamenané v súbore.
    - Poradové číslo rámca v analyzovanom súbore.
    - Dĺžku rámca v bajtoch poskytnutú pcap API, ako aj dĺžku tohto rámca prenášaného po médiu.
    - Typ rámca – Ethernet II, IEEE 802.3 (IEEE 802.3 s LLC, IEEE 802.3 s LLC a SNAP, IEEE 802.3 – Raw).
    - Zdrojovú a cieľovú fyzickú (MAC) adresu uzlov, medzi ktorými je rámec prenášaný.
- Výpis vnorených protokolov pre rámce typu Ethernet II a IEEE 802.3
- Analýzu IP adries a počtu odoslaných packetov
- Analýzu komunikácií zadaných protokolov

Implementácia a použité knižnice

Program je implementovaný v jazyku Python verzie 3.9 s využitím knižnice Scapy, z ktorej využíva funkciu na načítanie .pcap súboru *rdpcap(pathname)*. Na nasledovnú analýzu jednotlivých polí a hlavičiek rámcov nevyužíva iné funkcie poskytnuté knižnicou alebo programovacím jazykom.

Načítanie externého súboru

Program je organizovaný tak, že 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 sú programom načítané z jedného externého textového súboru.

Názvy a čísla portov sa pri spustení načítajú funkciou *nacitajSubor(...)* do slovníkov, kde sú uložené po celý beh programu. Ku každému názvu v súbore začínajúcemu znakom # patrí jeden slovník.

Ukážka externého súboru:

```
#Ethertypes
0800 IPv4
0806 ARP
0842 Wake-on-LAN
22f0 AVTP
86dd IPv6
88cc Link Layer Discovery Protocol
9000 Loopback
#LSAPs
42 STP
aa SNAP
e0 IPX
ff IPX
#IPProtocolNumbers
01 ICMP
02 IGMP
06 TCP
11 UDP
67 PIM
3a ICMPv6
#TCPs
0007 ECHO
```

```
0013 CHARGEN
0014 FTP-DATA
0015 FTP-CONTROL
0016 SSH
0017 TELNET
0050 HTTP
01bb HTTPs
#UDPs
0035 DNS
0045 TFTP
0208 Route Info Protocol
0089 NetBIOS Name Service
008a NetBIOS Datagram Service
076c Simple Service Discovery Protocol
14eb Link-local Multicast Name Resolution
#ICMPs
00 Echo Reply
03 Destination Unreachable
04 Source Quench
05 Redirect
08 Echo Request
09 Router Advertisement
0a Router Selection"
0b Time Exceeded
0c Parameter Problem
0d Timestamp
0e Timestamp Reply
0f Information Request
10 Information Reply
11 Address Mask Request
12 Address Mask Reply
1e Traceroute
```

Používateľské rozhranie

Používateľské rozhranie funguje v rámci konzoly, program sa ovláda vstupmi z klávesnice. Menu pozostáva z dvoch hlavných možností – prvej časti (body zadania 1-3) a druhej (bod 4). Okrem toho je možné vypísať obsah externého súboru (resp. už naplnených slovníkov) a tiež ukončiť program.

Po zvolení funkcie si program buď vypýta ďalšie upresnenie, alebo názov .pcap súboru, ktorý sa má spracovať. Program tiež kontroluje správnosť vstupov (názov súboru aj písmeno označujúce pokyn), a pri nesprávnom vstupe si vyžiada opätovné zadanie informácie.

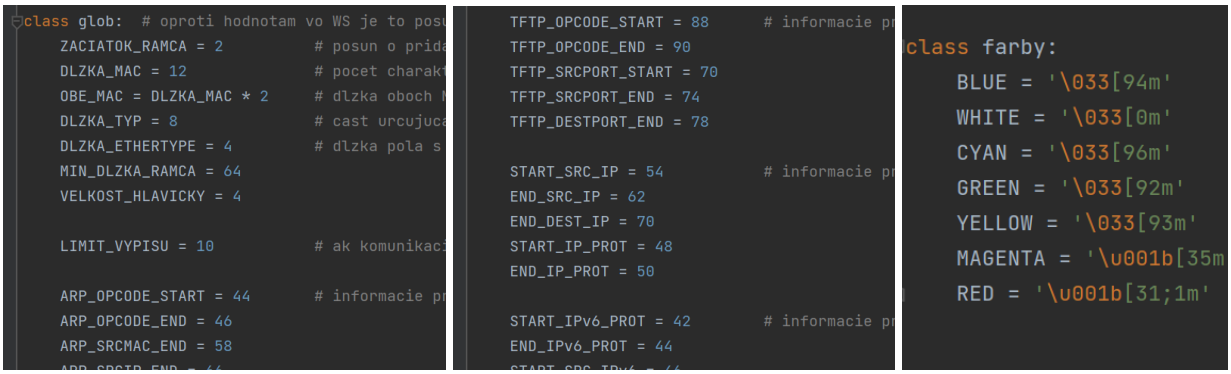
```
MENU
    -> pre prvu cast zadania stlac       (a)
    -> pre druhu cast zadania stlac      (b)
    -> pre vypis externeho suboru stlac  (c)
    -> pre ukoncenie programu stlac      (x)
```

```
b
    -> pre TFTP stlac    (g)
    -> pre ICMP stlac    (h)
    -> pre ARP stlac     (i)
    -> exit do menu      (x)

g

Zadaj nazov suboru: x

Subor x neexistuje
```

Globálne (statické) premenné - main

Program pre ľahkú prehľadnosť a zmeniteľnosť (vyhnutie sa priamemu umiestňovaniu čísiel do kódu) využíva globálne premenné v triede *glob*, v ktorých sú uložené začiatočné (a konečné) pozície analyzovaných polí jednotlivých typov rámcov.

Taktiež pre lepšiu čitateľnosť pracuje program s triedou farby (farebné odlíšenie výpisov).



```
class glob:  # oproti hodnotam vo WS je to posu
    ZACIATOK_RAMCA = 2          # posun o prida
    DLZKA_MAC = 12              # pocet charakt
    OBE_MAC = DLZKA_MAC * 2     # dlzka oboch M
    DLZKA_TYP = 8               # cast urcujuca
    DLZKA_ETHERTYPE = 4         # dlzka pola s
    MIN_DLZKA_RAMCA = 64
    VELKOST_HLAVICKY = 4

    LIMIT_VYPISU = 10           # ak komunikaci

    ARP_OPCODE_START = 44       # informacie pr
    ARP_OPCODE_END = 46
    ARP_SRCMAC_END = 58
```

```
    TFTP_OPCODE_START = 88      # informacie pr
    TFTP_OPCODE_END = 90
    TFTP_SRCPORT_START = 70
    TFTP_SRCPORT_END = 74
    TFTP_DESTPORT_END = 78

    START_SRC_IP = 54           # informacie pr
    END_SRC_IP = 62
    END_DEST_IP = 70
    START_IP_PROT = 48
    END_IP_PROT = 50

    START_IPv6_PROT = 42        # informacie pr
    END_IPv6_PROT = 44
```

```
class farby:
    BLUE = '\033[94m'
    WHITE = '\033[0m'
    CYAN = '\033[96m'
    GREEN = '\033[92m'
    YELLOW = '\033[93m'
    MAGENTA = '\u001b[35m
    RED = '\u001b[31;1m'
```

Bod 1 – výpis všetkých rámcov

Súbor vypíše všetky rámce v hexadecimálnom tvare postupne tak, ako boli zaznamenané v súbore. Pre každý rámec uvedie:

- poradové číslo rámca v analyzovanom súbore,
- dĺžku rámca v bajtoch poskytnutú pcap API, ako aj dĺžku tohto rámca prenášaného po médiu,
- typ rámca,
- zdrojovú a cieľovú fyzickú (MAC) adresu uzlov, medzi ktorými je rámec prenášaný.

Táto časť programu je sústredená v súbore *zadanie1.py*, a využíva najmä funkcie *start(...), get_typ_ramca(...), get_src_mac(...), get_dest_mac(...), print_ramec(...) a pkt_print(...)*, ktoré som implementovala.

Vo funkcii start sa program pokúsi otvoriť súbor – ak sa mu to podarí prečíta jeho dĺžku (počet rámcov v súbore), a pre každý z rámcov spustí funkciu *pkt_print*. Po vykonaní všetkých iterácií spustí funkciu pre IP štatistiku (viď. Bod 3).

Funkcia *pkt_print* postupne spracuje jednotlivé rámce súboru (vždy po jednom). Vypíše poradové číslo rámca (ktoré dostáva ako argument funkcie), vypočíta a vypíše dĺžku rámca (dĺžku API ako len(ramca), a dĺžku prenášanú po médiu buď ako minimálnu dĺžku 64 B, alebo dĺžku API spolu s hlavičkou 4 B).

Následne volanou funkciou *get_typ_ramca* program zistí, či je rámec typu Ethernet II alebo niektorý z typov IEEE 802.3. Funkciami *get_src_mac* a *get_dest_mac* zistí MAC adresy (podľa príslušných polí) a funkciou *getVnorenyProtokol* zistí vnorené protokoly a IP adresy (viď. Bod 2).

Ako posledné vypíše celý prenášaný rámec funkciou *print_ramec.*

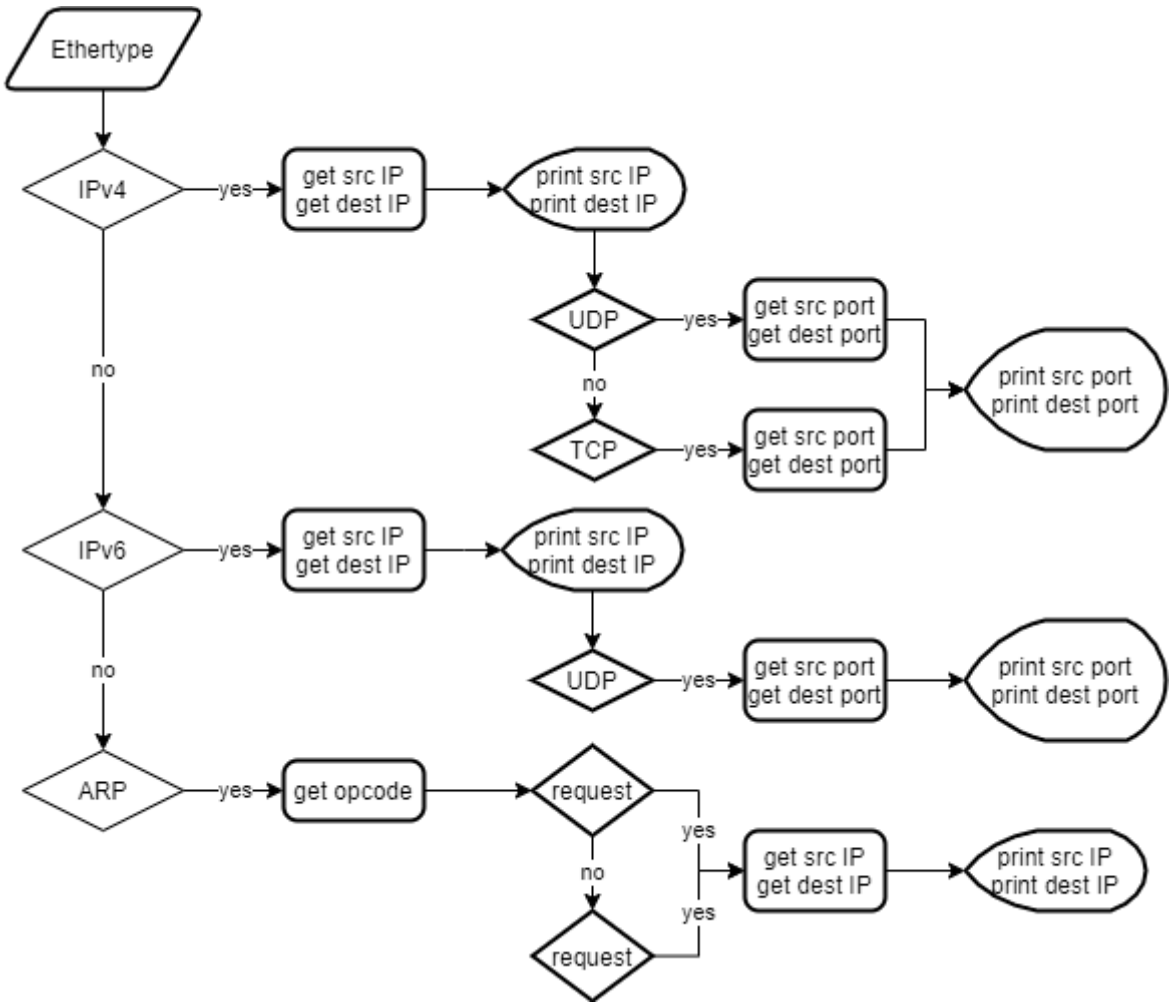
Bod 2 – výpis vnorených protokolov

Funkcie pre výpis vnorených protokolov sú v súbore zadanie2_3.py, a to napríklad funkcie *ipv4_getSrcIp(...), ipv6_getSrcIp(...)* alebo *getVnorenyProtokol(...).*

Tieto funkcie sú volané v predošlom bode funkciou *pkt_print* pre individuálne rámce súboru.

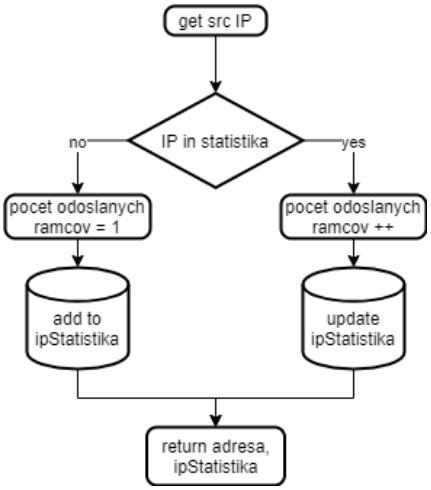
Funkcia *getVnorenyProtokol* dostane ako argument typ rámca (zisťovaný už v Bode 1 – Ethernet II alebo IEEE 802.3), a na základe toho sa pozrie na príslušné pole v rámci (pole Ethertype pre Ethernet II, a polia DSAP a SSAP pre IEEE 802.3). Prečítané informácie porovná so slovníkom obsahujúcim čísla a názvy protokolov – v prípade zhody vráti názov protokolu, inak vráti číslo/označenie portu.

Podľa zistenej informácie o vnorenom protokole ďalej analyzuje rámec – pre IPv4, IPv6 a ARP. Pre rámce typu IPv4 a IPv6 skúma zdrojové a cieľové IP adresy funkciami *getSrcIp* a *getDestIp* (ktoré sú rozdielne podľa toho, či ide o IPv4 alebo IPv6) a skúma tiež protokol UDP (a TCP pre IPv4) funkciami *ipv4_getVnorenyProtokol* a *ipv6_getVnorenyProtokol*. Pri ARP zisťuje, či ide o typ Request alebo Reply, a taktiež skúma IP adresy.



Bod 3 – analýza protokolov rodiny TCP/IPv4

Pri využívaní funkcie *ipv4_getSrcIp* v Bodoch 1 a 2 sa zaznamenané IPv4 adresy uložia do slovníka *ipStatistika* spolu s počtom rámcov, ktoré odoslali. Ak daná IPv4 adresa v slovníku nie je, uloží sa s počtom rámcov 1. Ak sa už v slovníku nachádza, počet rámcov sa inkrementuje.



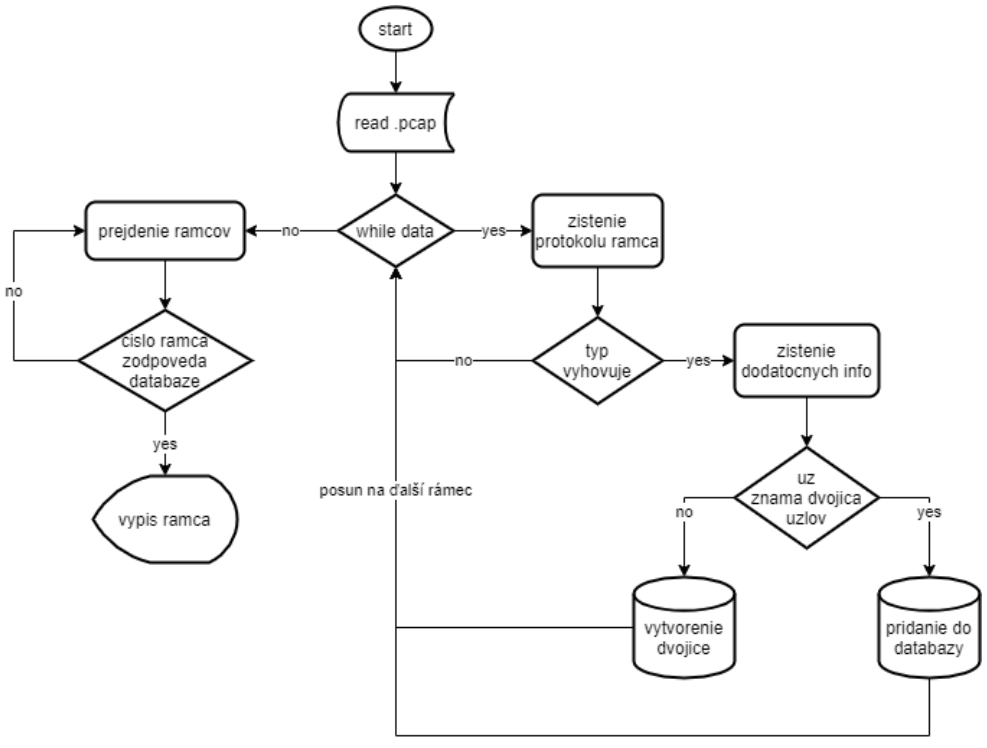
Funkcia *vypisIPstatistuku(...)* následne prejde slovník *ipStatistika* a vypíše všetky kľuče (IPv4 adresy odosielajúcich uzlov). Podľa hodnôt týchto kľúčov vie určiť IP adresu uzla, ktorý sumárne bez ohľadu na prijímateľa odoslal najväčší počet paketov.

Bod 4 – všeobecné informácie

Funkcie k tomuto bodu sú sústredené v súbore zadanie4.py. Obsahuje pomocné funkcie na prevod medzi sústavami (resp. na formátovanie adries), funkcie na finálne výpisy pre už analyzovaný súbor, a dve hlavné funkcie – *start* a *vyberProtokol*.

Funkcia *start* slúži obdobne ako pri bodoch 1, 2 a 3. Program načíta súbor a postupne pre každý rámec volá funkciu *vyberProtokol*, ktorá daný rámec analyzuje. Výstupom tejto analýzy je slovník, ktorý obsahuje informácie o komunikáciach – komunikujúce dvojice (adresy) ako key, a zoznam poradových čísiel rámcov so správnym typom komunikácie ako value.

Na koniec sa volá funkcia pre výpis (podľa zadaného vstupu), ktorá vypíše rámce zodpovedajúce poradovému číslu v slovníku.



Bod 4.a až 4.f – analýza TCP

Analýza všetkých protokolov rodiny TCP prebieha rovnako.

Vo funkcii *vyberProtokol* sa postupne prejdú všetky rámce, a tie, ktorých protokol sa zhoduje s tým hľadaným sa ďalej analyzujú.

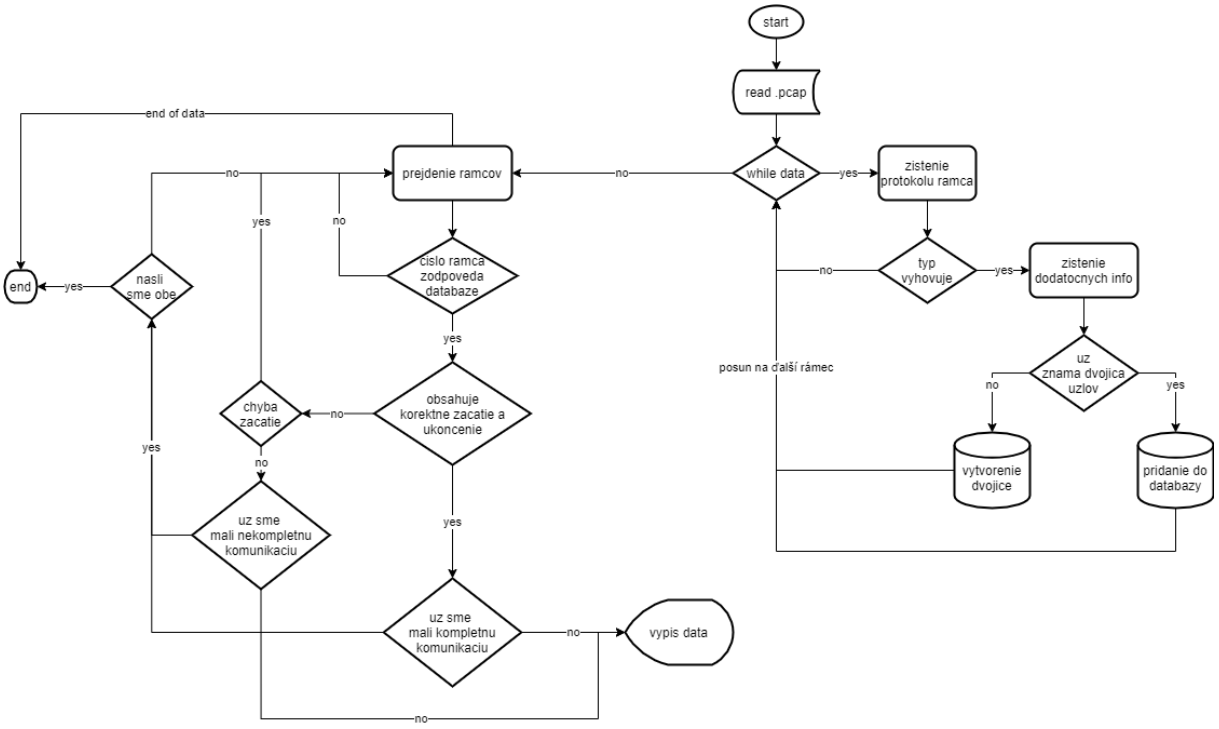
Podľa IP adries a portov komunikujúcich uzlov sa vytvorí ID. To potom program hľadá medzi už známymi uzlami. Ak ho nenájde, vytvorí nový záznam. Ak ho nájde, pridá číslo rámca do databázy.

Po vytvorení databázy dochádza ešte k filtrovaniu kompletných a nekompletných komunikácií vo funckií *vypisTCP*. Táto funkcia má zo všetkých záznamov vypísať iba prvú kompletnú a prvú nekompletnú komunikáciu.

Zo zadania, kompletná komunikácia je začatá otvorením SYN na oboch stranách, a ukončená FIN na oboch stranách, FIN a RST, alebo iba RST. Nekompletná neobsahuje ukončenie. Kompletná aj nekompletná komunikácia musí byť pritom začatá korektne.

Funkcia v rámcoch komunikácie skontroluje začiatok a hľadá ukončenie. Podľa toho, čo nájde (resp. nenájde) nastaví flagy, ktoré kontrolujú výpis.

Ak sa v súbore nenájde kompletná alebo nekompletná komunikácia, program o tom informuje.



Bod 4.g – analýza protokolov TFTP

Pre analýzu protokolov TFTP využívam tieto globálne premenné v triede *kom* v súbore *zadanie4.py*:

```
tftp = {}           # pre TFTP
tftp_no = 0
opcode = ""
prvy_opcode = ""
tftp_a_ip = ""
tftp_b_ip = ""
tftp_a_port = ""
tftp_b_port = ""
```

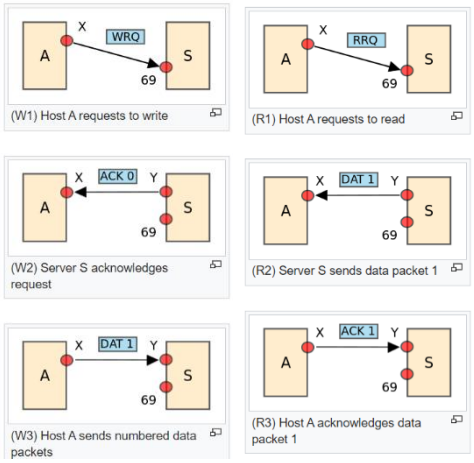
- Slovník *tftp*, v ktorom ukladám komunikujúce uzly (ako key) a zoznam obsahujúci poradové čísla rámcov, v ktorých som našla TFTP komunikáciu (ako value).
- Premenná *tftp_no* označuje priebežne poradové číslo TFTP komunikácie.
- Premenná *opcode* slúži ako história pre predchádzajúci opcode pred tým, ktorý sa aktuálne spracúva (opcode predchádzajúceho rámca TFTP komunikácie).
- Premenná *prvy_opcode* značí prvý opcode aktuálnej komunikácie.
- Premenné *tftp_a_ip* a *tftp_a_port* nesú informácie o uzly ktorý inicializoval komunikáciu, premenné *tftp_b_ip* a *tftp_b_port* zasa informácie o adresátovi.

Funkcia *vyberProtokol* zistí pomocou pomocných funkcií vnorený protokol rámca (IPv4 -> UDP -> TFTP), a ak nájde prvý rámec na port 69:

- inkrementuje poradové číslo TFTP komunikácie,
- zistí IP adresy komunikujúcich uzlov,
- vytvorí z týchto údajov ID pre slovník,
- nastaví hodnoty pre globálne premenné prvy_opcode, opcode a lokálnu premennú opcode,
- uloží hodnoty portov do globálnych premenných,
- a na záver pridá ID komunikácie a poradové číslo rámca do slovníka.

Ak nájde iný rámec UDP:

- zistí, či tento rámec patrí do TFTP komunikácie podľa IP adries a portov (až v druhej správe zistí port uzlu b, a ten uloží pre ďalšie kontrolovanie),
- vytvorí ID tak, aby sa komunikácie správne spárovali (správa z uzla a -> b patrí do tej istej komunikácie ako z b -> a),
- skontroluje, či je správna následnosť opkódov (ako na obrázku, prípadne error code),
- aktualizuje slovník – pre dané ID pridá do zoznamu poradové číslo rámca.



Na záver (po tom ako sa zanalyzujú všetky rámce a priradia do komunikácie) sa správne rámce vypíšu pomocou funkcie *vypisTFTP*.

Bod 4.h – analýza ICMP správ

Pre analýzu protokolov ICMP využívam globálnu premennú - slovník *icmp* - v triede *kom* v súbore *zadanie4.py*. Slovník *icmp* ukladá komunikujúce uzly (ako key) a zoznam obsahujúci poradové čísla rámcov, v ktorých som našla ICMP komunikáciu (ako value).

Funkcia *vyberProtokol* zistí pomocou pomocných funkcií vnorený protokol rámca (IP -> ICMP), a ak nájde takýto protokol:

- zistí IP adresy,
- vytvorí z nich ID,
- vytvorí nový záznam v slovníku alebo aktualizuje slovník o poradové číslo rámca pre danú dvojicu uzlov.

Vo výpise (funkcia *vypisICMP*) sa okrem iného zisťuje typ ICMP správy (Echo Request, Echo Reply...). Tieto hodnoty má program z externého súboru.

Bod 4.i – analýza protokolov ARP

Pre analýzu protokolov ARP využívam tieto globálne premenné v triede *kom* v súbore *zadanie4.py*:

```
class kom:
    arp = {}                # pre ARP
    arp_komp_no = 1
```

- Slovník *arp*, v ktorom ukladám komunikujúce uzly (ako key) a zoznam obsahujúci poradové čísla rámcov, v ktorých som našla TFTP komunikáciu (ako value).
- Premennú *arp_komp_no*, ktorá uchováva počet kompletných komunikácií (poradové číslo kompletnej komunikácie sa využíva pre párovanie komunikácií)

Funkcia *vyberProtokol* zistí pomocou pomocných funkcií vnorený protokol rámca (Ethernet -> ARP), a ak nájde ARP protokol:

- zistí opcode a IP adresy,
- vytvorí z nich ID,
- ak dvojicu nepozná:
    - vytvorí nový záznam,
- ak dvojicu pozná:
    - ak ide o Reply a našli sme kompletnú komunikáciu
        - pridá rámec ako posledný pre danú komunikáciu uzlov
        - zmení ID uzavretej komunikácie, aby sa do nej nepridávali ďalšie prípadné správy medzi uzlami
    - ak je to ďalší Request pridá ho k predchádzajúcim

Po vykonaní tejto funkcie sa na záver volá výpis – *vypisARP*.

Ukážky výpisu

Bod 1 až 3

Ethernet II – IPv4

```
ramec 23
dlzka ramca poskytnuta pcap API - 70 B
dlzka ramca prenasaneho po mediu - 74 B
Ethernet II
Zdrojova MAC adresa: CC 08 09 D4 00 00
Cielova MAC adresa:  02 00 4C 4F 4F 50
IPv4
zdrojova IP adresa: 12.0.0.1
cielova IP adresa:  12.0.0.5
UDP
dest port: 69 - TFTP
src port:  49917
02 00 4C 4F 4F 50 CC 08   09 D4 00 00 08 00 45 00
00 38 00 00 00 00 FF 11   A3 AF 0C 00 00 01 0C 00
00 05 C2 FD 00 45 00 24   86 3B 00 01 63 6D 65 2D
67 75 69 2D 34 2E 31 2E   30 2E 31 2E 74 61 72 00
6F 63 74 65 74 00
```

```
ramec 42
dlzka ramca poskytnuta pcap API - 60 B
dlzka ramca prenasaneho po mediu - 64 B
Ethernet II
Zdrojova MAC adresa: 84 B8 02 66 72 34
Cielova MAC adresa:  30 65 EC 83 32 36
IPv4
zdrojova IP adresa: 201.43.83.60
cielova IP adresa:  147.175.145.165
TCP
src port:  13445
dest port: 8080
30 65 EC 83 32 36 84 B8   02 66 72 34 08 00 45 00
00 28 E9 8E 40 00 F0 06   5F 84 C9 2B 53 3C 93 AF
91 A5 34 85 1F 90 B7 84   0F F3 00 00 00 00 50 02
39 08 19 91 00 00 00 00   12 8B 2B 21
```

```
ramec 2
dlzka ramca poskytnuta pcap API - 114 B
dlzka ramca prenasaneho po mediu - 118 B
Ethernet II
Zdrojova MAC adresa: CC 08 09 D4 00 00
Cielova MAC adresa:  02 00 4C 4F 4F 50
IPv4
zdrojova IP adresa: 12.0.0.1
cielova IP adresa:  12.0.0.5
ICMP
Echo Request
02 00 4C 4F 4F 50 CC 08   09 D4 00 00 08 00 45 00
00 64 00 08 00 00 FF 01   A3 8B 0C 00 00 01 0C 00
00 05 08 00 98 B2 00 02   00 00 00 00 00 00 00 01
E5 94 AB CD AB CD AB CD   AB CD AB CD AB CD AB CD
AB CD AB CD AB CD AB CD   AB CD AB CD AB CD AB CD
AB CD AB CD AB CD AB CD   AB CD AB CD AB CD AB CD
AB CD AB CD AB CD AB CD   AB CD AB CD AB CD AB CD
AB CD
```

Ethernet II – IPv6

```
ramec 604
dlzka ramca poskytnuta pcap API - 208 B
dlzka ramca prenasaneho po mediu - 212 B
Ethernet II
Zdrojova MAC adresa: EC B1 D7 97 25 9B
Cielova MAC adresa:  33 33 00 00 00 0C
IPv6
zdrojova adresa: fe80:0000:0000:0000:1962:13e9:3505:dcec
cielova adresa:  ff02:0000:0000:0000:0000:0000:000c
UDP
dest port: 1900 - Simple Service Discovery Protocol
src port:  57311
33 33 00 00 00 0C EC B1   D7 97 25 9B 86 DD 60 00
00 00 00 9A 11 01 FE 80   00 00 00 00 00 00 19 62
13 E9 35 05 DC EC FF 02   00 00 00 00 00 00 00 00
00 00 00 00 00 0C DF DF   07 6C 00 9A 78 42 4D 2D
53 45 41 52 43 48 20 2A   20 48 54 54 50 2F 31 2E
31 0D 0A 48 6F 73 74 3A   5B 46 46 30 32 3A 3A 43
5D 3A 31 39 30 30 0D 0A   53 54 3A 75 72 6E 3A 4D
69 63 72 6F 73 6F 66 74   20 57 69 6E 64 6F 77 73
20 50 65 65 72 20 4E 61   6D 65 20 52 65 73 6F 6C
75 74 69 6F 6E 20 50 72   6F 74 6F 63 6F 6C 3A 20
56 34 3A 49 50 56 36 3A   4C 69 6E 6B 4C 6F 63 61
6C 0D 0A 4D 61 6E 3A 22   73 73 64 70 3A 64 69 73
63 6F 76 65 72 22 0D 0A   4D 58 3A 33 0D 0A 0D 0A
```

```
ramec 7
dlzka ramca poskytnuta pcap API - 148 B
dlzka ramca prenasaneho po mediu - 152 B
Ethernet II
Zdrojova MAC adresa: 94 DE 80 43 1A CB
Cielova MAC adresa:  33 33 00 01 00 02
IPv6
zdrojova adresa: fe80:0000:0000:0000:d121:7eb1:887d:d1dc
cielova adresa:  ff02:0000:0000:0000:0000:0001:0002
UDP
src port:  546
dest port: 547
33 33 00 01 00 02 94 DE   80 43 1A CB 86 DD 60 00
00 00 00 5E 11 01 FE 80   00 00 00 00 00 00 D1 21
7E B1 88 7D D1 DC FF 02   00 00 00 00 00 00 00 00
00 00 00 01 00 02 02 22   02 23 00 5E 76 2A 01 EE
95 DE 00 08 00 02 0C 1C   00 01 00 0E 00 01 00 01
19 79 65 BE 94 DE 80 43   1A CB 00 03 00 0C 11 94
DE 80 00 00 00 00 00 00   00 00 00 27 00 08 00 06
50 43 2D 32 38 39 00 10   00 0E 00 00 01 37 00 08
4D 53 46 54 20 35 2E 30   00 06 00 08 00 18 00 17
00 11 00 27
```

Ethernet II – ARP

```
ramec 605
dlzka ramca poskytnuta pcap API - 60 B
dlzka ramca prenasaneho po mediu - 64 B
Ethernet II
Zdrojova MAC adresa: 84 B8 02 66 72 34
Cielova MAC adresa:  FF FF FF FF FF FF
ARP
Type: Request
zdrojova IP adresa: 147.175.144.1
cielova IP adresa:  147.175.145.139
FF FF FF FF FF FF 84 B8   02 66 72 34 08 06 00 01
08 00 06 04 00 01 84 B8   02 66 72 34 93 AF 90 01
00 00 00 00 00 00 93 AF   91 8B 00 00 00 00 00 00
00 00 00 00 00 00 00 00   5E C2 7D 24
```

```
ramec 1207
dlzka ramca poskytnuta pcap API - 60 B
dlzka ramca prenasaneho po mediu - 64 B
Ethernet II
Zdrojova MAC adresa: 84 B8 02 66 72 34
Cielova MAC adresa:  30 65 EC 83 32 36
ARP
Type: Reply
zdrojova IP adresa: 147.175.144.1
cielova IP adresa:  147.175.145.165
30 65 EC 83 32 36 84 B8   02 66 72 34 08 06 00 01
08 00 06 04 00 02 84 B8   02 66 72 34 93 AF 90 01
30 65 EC 83 32 36 93 AF   91 A5 00 00 00 00 00 00
00 00 00 00 00 00 00 00   DE D6 9E 9D
```

Ethernet II – iné

```
ramec 325
dlzka ramca poskytnuta pcap API - 373 B
dlzka ramca prenasaneho po mediu - 377 B
Ethernet II
Zdrojova MAC adresa: B8 AF 67 DE C4 ED
Cielova MAC adresa:  01 80 C2 00 00 0E
Link Layer Discovery Protocol
01 80 C2 00 00 0E B8 AF   67 DE C4 ED 88 CC 02 07
04 B8 AF 67 DF 28 16 04   16 05 47 69 67 61 62 69
74 45 74 68 65 72 6E 65   74 35 2F 30 2F 33 33 06
02 00 78 08 1F 47 69 67   61 62 69 74 45 74 68 65
72 6E 65 74 35 2F 30 2F   33 33 20 49 6E 74 65 72
66 61 63 65 0A 02 48 50   0C BA 48 50 20 43 6F 6D
77 61 72 65 20 50 6C 61   74 66 6F 72 6D 20 53 6F
66 74 77 61 72 65 2C 20   53 6F 66 74 77 61 72 65
20 56 65 72 73 69 6F 6E   20 35 2E 32 30 20 52 65
6C 65 61 73 65 20 32 32   30 38 50 30 31 0D 0A 48
50 20 41 35 35 30 30 2D   34 38 47 2D 50 6F 45 2B
20 45 49 20 53 77 69 74   63 68 20 77 69 74 68 20
32 20 49 6E 74 65 72 66   61 63 65 20 53 6C 6F 74
73 0D 0A 43 6F 70 79 72   69 67 68 74 20 28 63 29
20 32 30 31 30 2D 32 30   31 31 20 48 65 77 6C 65
74 74 2D 50 61 63 6B 61   72 64 20 44 65 76 65 6C
6F 70 6D 65 6E 74 20 43   6F 6D 70 61 6E 79 2C 20
4C 2E 50 2E 0E 04 00 14   00 14 10 0D 05 01 93 AF
90 02 02 00 00 00 36 01   00 FE 06 00 80 C2 01 00
01 FE 07 00 80 C2 02 02   00 00 FE 10 00 80 C2 03
00 01 09 56 4C 41 4E 20   30 30 30 31 FE 09 00 12
0F 01 03 00 00 00 1E FE   07 00 12 0F 02 03 01 01
FE 09 00 12 0F 03 01 00   00 00 00 FE 06 00 12 0F
04 24 00 00 00
```

IEEE 802.3

```
ramec 544
dlzka ramca poskytnuta pcap API - 64 B
dlzka ramca prenasaneho po mediu - 68 B
IEEE 802.3 s LLC a SNAP
Zdrojova MAC adresa: A8 97 DC 9B AD 17
Cielova MAC adresa:  01 00 0C CC CC CD
SNAP
01 00 0C CC CC CD A8 97   DC 9B AD 17 00 32 AA AA
03 00 00 0C 01 0B 00 00   02 02 7E 80 5D A8 97 DC
9B AD 00 00 00 00 00 80   5D A8 97 DC 9B AD 00 80
17 00 00 14 00 02 00 0F   00 00 00 00 00 02 00 D5
```

```
ramec 1293
dlzka ramca poskytnuta pcap API - 110 B
dlzka ramca prenasaneho po mediu - 114 B
IEEE 802.3 - Raw
Zdrojova MAC adresa: 00 17 08 87 9A 1F
Cielova MAC adresa:  FF FF FF FF FF FF
IPX
FF FF FF FF FF FF 00 17   08 87 9A 1F 00 60 FF FF
00 60 00 04 30 09 80 00   FF FF FF FF FF FF 04 52
30 09 80 00 00 17 08 87   9A 1F 04 52 00 02 03 0C
30 30 31 37 30 38 38 37   39 41 31 46 38 33 44 5A
4E 50 49 38 37 39 41 31   46 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00   00 00 00 00 00 00 00 00
30 09 80 00 00 17 08 87   9A 1F 40 0C 00 01
```

Bod 4

TCP komunikacie

```
Vypis HTTP komunikacii

Komunikacia c. 1
Prva kompletna komunikacia, pocet ramcov v komunikacii: 10

ramec 130
flag: SYN
dlzka ramca poskytnuta pcap API - 62 B
dlzka ramca prenasaneho po mediu - 66 B
Ethernet II
Zdrojova MAC adresa: 00 14 38 06 E0 93
Cielova MAC adresa:  00 02 CF AB A2 4C
TCP
zdrojova IP adresa: 192.168.1.33
cielova IP adresa:  147.175.1.55
dest port: 80 - HTTP
src port:  3980
00 02 CF AB A2 4C 00 14   38 06 E0 93 08 00 45 00
00 30 16 18 40 00 80 06   8E 00 C0 A8 01 21 93 AF
01 37 0F 8C 00 50 8F 64   D4 93 00 00 00 00 70 02
FF FF B8 9B 00 00 02 04   05 B4 01 01 04 02

ramec 131
flag: SYN ACK
dlzka ramca poskytnuta pcap API - 62 B
dlzka ramca prenasaneho po mediu - 66 B
Ethernet II
Zdrojova MAC adresa: 00 02 CF AB A2 4C
Cielova MAC adresa:  00 14 38 06 E0 93
TCP
zdrojova IP adresa: 147.175.1.55
cielova IP adresa:  192.168.1.33
src port:  80 - HTTP
dest port: 3980
00 14 38 06 E0 93 00 02   CF AB A2 4C 08 00 45 00
00 30 00 00 40 00 3A 06   EA 18 93 AF 01 37 C0 A8
01 21 00 50 0F 8C F3 73   9C D8 8F 64 D4 94 70 12
16 D0 11 BE 00 00 02 04   05 64 01 01 04 02

ramec 132
flag: ACK
dlzka ramca poskytnuta pcap API - 54 B
dlzka ramca prenasaneho po mediu - 64 B
Ethernet II
Zdrojova MAC adresa: 00 14 38 06 E0 93
Cielova MAC adresa:  00 02 CF AB A2 4C
TCP
zdrojova IP adresa: 192.168.1.33
cielova IP adresa:  147.175.1.55
```

TFTP komunikácie

```
Error Code,
od  12.0.0.1 port 51267
pre 12.0.0.5 port 1516
ramec 72
dlzka ramca poskytnuta pcap API - 65 B
dlzka ramca prenasaneho po mediu - 69 B
Ethernet II
Zdrojova MAC adresa: CC 08 09 D4 00 00
Cielova MAC adresa:  02 00 4C 4F 4F 50
IPv4
zdrojova IP adresa: 12.0.0.1
cielova IP adresa:  12.0.0.5
UDP
02 00 4C 4F 4F 50 CC 08   09 D4 00 00 08 00 45 00
00 33 00 09 00 00 FF 11   A3 AB 0C 00 00 01 0C 00
00 05 C8 43 05 EC 00 1F   63 F3 00 05 00 00 53 65
73 73 69 6F 6E 20 74 65   72 6D 69 6E 61 74 65 64
00

Komunikacia c. 4
pocet ramcov v komunikacii: 3163

Read Request,
od  12.0.0.1 port 50304
pre 12.0.0.5 port 69
ramec 75
dlzka ramca poskytnuta pcap API - 70 B
dlzka ramca prenasaneho po mediu - 74 B
Ethernet II
Zdrojova MAC adresa: CC 08 09 D4 00 00
Cielova MAC adresa:  02 00 4C 4F 4F 50
IPv4
zdrojova IP adresa: 12.0.0.1
cielova IP adresa:  12.0.0.5
UDP
02 00 4C 4F 4F 50 CC 08   09 D4 00 00 08 00 45 00
00 38 00 00 00 00 FF 11   A3 AF 0C 00 00 01 0C 00
00 05 C4 80 00 45 00 24   84 B8 00 01 63 6D 65 2D
67 75 69 2D 34 2E 31 2E   30 2E 31 2E 74 61 72 00
6F 63 74 65 74 00

Data,
od  12.0.0.5 port 1517
pre 12.0.0.1 port 50304
ramec 76
dlzka ramca poskytnuta pcap API - 558 B
dlzka ramca prenasaneho po mediu - 562 B
Ethernet II
Zdrojova MAC adresa: 02 00 4C 4F 4F 50
Cielova MAC adresa:  CC 08 09 D4 00 00
IPv4
zdrojova IP adresa: 12.0.0.5
cielova IP adresa:  12.0.0.1
```

ICMP komunikácie

```
VYPIS ICMP KOMUNIKACII

Komunikacia c. 1
pocet ramcov v komunikacii: 32

ramec 2
Echo Request
dlzka ramca poskytnuta pcap API - 110 B
dlzka ramca prenasaneho po mediu - 114 B
Ethernet II
Zdrojova MAC adresa: 00 14 38 06 E0 93
Cielova MAC adresa:  00 02 CF AB A2 4C
IPv4
zdrojova IP adresa: 192.168.1.33
cielova IP adresa:  158.195.4.138
ICMP
00 02 CF AB A2 4C 00 14   38 06 E0 93 08 00 4E 00
00 60 BB 9C 00 00 80 01   C7 C4 C0 A8 01 21 9E C3
04 8A 44 24 05 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 08 00   36 5C 03 00 14 00 61 62
63 64 65 66 67 68 69 6A   6B 6C 6D 6E 6F 70 71 72
73 74 75 76 77 61 62 63   64 65 66 67 68 69

ramec 3
Echo Reply
dlzka ramca poskytnuta pcap API - 110 B
dlzka ramca prenasaneho po mediu - 114 B
Ethernet II
Zdrojova MAC adresa: 00 02 CF AB A2 4C
Cielova MAC adresa:  00 14 38 06 E0 93
IPv4
zdrojova IP adresa: 158.195.4.138
cielova IP adresa:  192.168.1.33
ICMP
00 14 38 06 E0 93 00 02   CF AB A2 4C 08 00 4E 00
00 60 C2 2E 00 00 37 01   C8 43 9E C3 04 8A C0 A8
01 21 44 24 25 81 C3 1C   4B 81 05 02 91 43 C0 6C
94 4B 05 02 91 42 C1 57   03 89 05 02 91 44 9E C3
01 5D 05 02 91 44 00 00   3E 5C 03 00 14 00 61 62
63 64 65 66 67 68 69 6A   6B 6C 6D 6E 6F 70 71 72
```

ARP komunikácie

```
Komunikacia c. 166
pocet ramcov v komunikacii: 2

ARP-Request, IP adresa: 147.175.144.1 MAC adresa: 84 B8 02 66 72 34
Zdrojova IP: 147.175.145.165 Cielova IP adresa: 147.175.144.1
ramec 1206
dlzka ramca poskytnuta pcap API - 42 B
dlzka ramca prenasaneho po mediu - 64 B
Ethernet II
ARP
Zdrojova MAC adresa: 30 65 EC 83 32 36
Cielova MAC adresa:  84 B8 02 66 72 34
84 B8 02 66 72 34 30 65   EC 83 32 36 08 06 00 01
08 00 06 04 00 01 30 65   EC 83 32 36 93 AF 91 A5
84 B8 02 66 72 34 93 AF   90 01

ARP-Reply, IP adresa: 147.175.144.1 MAC adresa: 84 B8 02 66 72 34
Zdrojova IP: 147.175.144.1 Cielova IP adresa: 147.175.145.165
ramec 1207
dlzka ramca poskytnuta pcap API - 60 B
dlzka ramca prenasaneho po mediu - 64 B
Ethernet II
ARP
Zdrojova MAC adresa: 84 B8 02 66 72 34
Cielova MAC adresa:  30 65 EC 83 32 36
30 65 EC 83 32 36 84 B8   02 66 72 34 08 06 00 01
08 00 06 04 00 02 84 B8   02 66 72 34 93 AF 90 01
30 65 EC 83 32 36 93 AF   91 A5 00 00 00 00 00 00
00 00 00 00 00 00 00 00   DE D6 9E 9D

Komunikacia c. 167
pocet ramcov v komunikacii: 1

ARP-Request, IP adresa: 147.175.144.42 MAC adresa: ???
Zdrojova IP: 147.175.145.63 Cielova IP adresa: 147.175.144.42
ramec 1210
dlzka ramca poskytnuta pcap API - 60 B
dlzka ramca prenasaneho po mediu - 64 B
Ethernet II
ARP
Zdrojova MAC adresa: 00 16 17 E2 9B F4
Cielova MAC adresa:  FF FF FF FF FF FF
FF FF FF FF FF FF 00 16   17 E2 9B F4 08 06 00 01
08 00 06 04 00 01 00 16   17 E2 9B F4 93 AF 91 3F
00 00 00 00 00 00 93 AF   90 2A 00 00 00 00 00 00
00 00 00 00 00 00 00 00   00 00 00 00
```