

VYSOKÉ UČENIE TECHNICKÉ V BRNE
FAKULTA INFORMAČNÝCH TECHNOLOGIÍ

Počítačové komunikácie a siete – 2. projekt
Varianta ZETA - Sniffer paketov

Obsah

1	Cieľ projektu a výber jazyka	2
2	Návod k použitiu	2
3	Spracovanie argumentov	2
3.1	Podporované parametre	2
3.2	Ošetrenie neplatných hodnôt argumentov	2
4	Rozdelenie kľúčových pojmov	2
5	Zachytávanie a filtrovanie paketov	3
5.1	Podporované typy	3
5.2	Implementácia zachytávania paketov	3
5.3	Výpis paketov	3
5.3.1	Výpis hlavičiek	3
5.3.2	Výpis dát	3
6	Testovanie	4
6.1	Testovanie TCP paketov – referenčný stroj	4
6.2	Testovanie UDP paketov – referenčný stroj	5
6.3	Testovanie ARP paketov – Windows	5
6.4	Testovanie ICMP paketov – Windows	6
6.4.1	Testovanie ICMPv4 – Windows	6
6.4.2	Testovanie ICMPv6 – Windows	7
7	Záver	8
8	Odkazy na referencie	9

1 Cieľ projektu a výber jazyka

Cielom projektu je vytvoriť sieťový analyzátor, ktorý zachytáva a filtruje pakety na sieťovom rozhraní. Ako implementačný jazyk bol zvolený jazyk C#. Projekt je rozdelený na dva zdrojové súbory `Program.cs` a `Sniffer.cs`.

2 Návod k použitiu

K riešeniu je priložený `Makefile`, ktorý pomocou príkazu `make` zostaví projekt. Ten je následne k dispozícii v adresári `/run`. Spustenie projektu na referenčnom virtuálnom stroji v adresári `/run` je možné pomocou príkazu `sudo ./ipk-sniffer [argumenty]`. Pre vyčistenie projektového adresára a zbavenie sa súborov vytvorených prekladom je možné použiť v koreňovom adresári príkaz `make clean`.

3 Spracovanie argumentov

Spracovanie argumentov sa nachádza v súbore `Program.cs`. O ošetrenie zadania nepodporovaných argumentov sa stará knižnica `System.Commandline`. [2] Pri nepodporovaných argumentoch sa ukončí činnosť programu a vypíše sa nápoveda, tá je k dispozícii aj pri spustení projektu s argumentom `-h` alebo `--help`.

3.1 Podporované parametre

`--interface | -i <rozhranie>` – určí rozhranie určené k naslúchaniu
`-p <port>` – filtrácia paketov pomocou portu, bez uvedenia sú akceptované všetky porty
`-n <počet>` – určuje počet paketov, ktoré analyzátor spracuje
`--tcp | -t` – zachytávanie TCP paketov
`--udp | -u` – zachytávanie UDP paketov
`--arp` – zachytávanie ARP paketov
`--icmp` – zachytávanie ICMP paketov
`--help | -h` – výpis nápovedy

3.2 Ošetrenie neplatných hodnôt argumentov

Pri zadaní nedostupného/neexistujúceho rozhrania, sa vypíše chybová hláška, zobrazí sa zoznam dostupných zariadení a ukončí sa vykonávanie programu. ¹ Pri porte mimo interval $< 0,65535 >$ sa ukončí činnosť programu. [8] Pri zadaní záporného počtu paketov ale aj pri nezadaní argumentu `-n` sa spracuje jeden paket.

4 Rozdelenie kľúčových pojmov

Transportná vrstva	TCP	UDP
Sieťová vrstva	IPv4	IPv6 ICMP
Linková vrstva	Ethernet	ARP

Pred samotným predstavením implementácie rozdelíme kľúčové pojmy zo sekcie 5 podľa jednotlivých vrstiev. [6]

¹V riešení sa využívajú dva návratové kódy - 0 pre úspech a 1 pre neúspech.

5 Zachytávanie a filtrovanie paketov

Pre zachytávanie paketov sa využívajú knižnice `SharpPcap` a `PacketDotNet`. [3] Pred zachytávaním prebieha kontrola na typ filtru zo vstupu a ak nie sú špecifikované žiadne typy paketov pre zachytávanie, tak sa nastaví všetky podporované typy za povolené. Počas zachytávania paketov sú už tieto filtry kontrolované iba jednotlivo.

5.1 Podporované typy

Riešenie podporuje zachytávanie TCP paketov, UDP paketov a ICMP paketov (všetky tieto pakety pre IPv4 aj IPv6). ARP protokol je definovaný len pre IPv4. [5] Pri ARP, ICMPv4 a ICMPv6 paketoch sa ignoruje vstupný argument `-p` reprezentujúci port.

5.2 Implementácia zachytávania paketov

Po otvorení rozhrania na počúvanie sa pakety získavajú pomocou metódy `GetNextPacket()` z knižnice `SharpPcap`. [1] Takto získaný paket obsahuje len záznam o čase, vrstve a dáta. Najskôr prebieha kontrola podľa linkovej vrstvy dátového spojenia, pričom podporovaná je len linková vrstva `Ethernet`. [6] Následne sa dáta reprezentované počtom bytov prevedú na typ `Packet` z knižnice `PacketDotNet`. Pre určenie typu jednotlivých paketov sa používa generická metóda `Extract`. Pre TCP a UDP [4] pakety sa táto metóda využíva pre určenie adres v závislosti na rodičovskom pakete zo sieťovej vrstvy, ktorá určuje či ide o IPv4 alebo IPv6. Pre ARP rámce [7] sa namiesto zdrojovej a cieľovej IP adresy s portami ukladá IP adresa zariadenia odosiateľa a príjemateľa. Pre ICMP pakety sa ukladajú zdrojové a cieľové adresy bez portov. Po spracovaní paketov sa ukončí spojenie s rozhraním a následne sa ukončí vykonávanie programu.

5.3 Výpis paketov

Výpis paketov rozdeľuje riešenie do dvoch častí. V prvej sa vypíše čas, IP adresy, dĺžka a pre určité pakety aj porty. Následne sa spracujú a vypíšu dáta paketu.

5.3.1 Výpis hlavičiek

Pred výpisom hlavičiek paketov sa spočíta časový posun aby výsledný formát pre čas podľa RFC3339. Následne sa vypíšu hlavičky pričom TCP a UDP pakety zdieľajú jednu metódu na výpis (pakety so zdrojovým a cieľovým portom). ARP a ICMP pakety zdieľajú druhú metódu (pakety bez portov).

5.3.2 Výpis dát

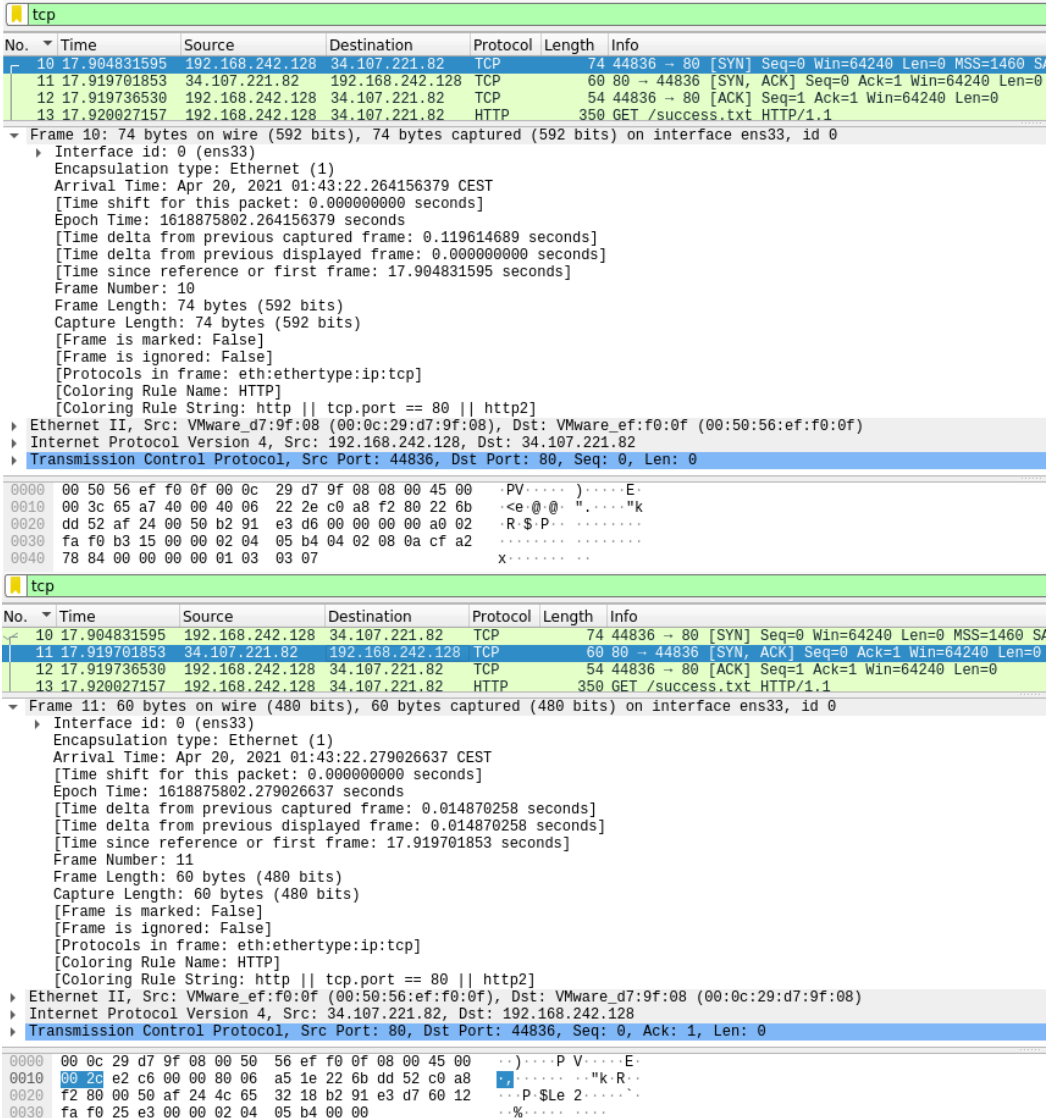
Pred výpisom dát sa najskôr dáta rozdelia po 16 bytoch a odošlú sa na výpis. Pri výpise sa najskôr vypočíta a vypíše posun, následne sú jednotlivé byty prevedené do hexadecimálnej sústavy a následne sú prevedené na znaky a vypísané podľa ASCII hodnoty. Pri výpise posledného riadku s menším počtom bytov ako 16 sa dopočíta zarovnanie pomocou vzorca $(16 - \text{dĺžka bytov riadku}) * 3$. V prípade menej ako polovičného zaplnenia posledného riadku sa k zarovnaniu pripočíta ďalšia medzera.

6 Testovanie

Pre účely testovania projektu a porovnanie som využil nástroj Wireshark. Testovanie prebehlo na osobnom stroji s Winowsovou distribúciou a tiež na referenčnom stroji s Unixovou distrúciou.

6.1 Testovanie TCP paketov – referenčný stroj

```
student@student-vm:~/ipk-sniffer/run$ sudo ./ipk-sniffer -i ens33 --tcp -n 2
2021-04-19T23:43:22.264+02:00 192.168.242.128 : 44836 > 34.107.221.82 : 80, length 74 bytes
0x0000: 00 50 56 ef f0 0f 00 0c 29 d7 9f 08 08 00 45 00 .PV....)....E.
0x0010: 00 3c 65 a7 40 00 40 06 22 2e c0 a8 f2 80 22 6b .<e.@.@. "...."k
0x0020: dd 52 af 24 00 50 b2 91 e3 d6 00 00 00 00 a0 02 .R.$P.. ....
0x0030: fa f0 b3 15 00 00 02 04 05 b4 04 02 08 0a cf a2 .....
0x0040: 78 84 00 00 00 00 01 03 03 07 X.....
2021-04-19T23:43:22.279+02:00 34.107.221.82 : 80 > 192.168.242.128 : 44836, length 60 bytes
0x0000: 00 0c 29 d7 9f 08 00 50 56 ef f0 0f 08 00 45 00 ..)....P V....E.
0x0010: 00 2c e2 c6 00 00 80 06 a5 1e 22 6b dd 52 c0 a8 .,....."k.R..
0x0020: f2 80 00 50 af 24 4c 65 32 18 b2 91 e3 d7 60 12 ...P.$Le 2.....
0x0030: fa f0 25 e3 00 00 02 04 05 b4 00 00 ..%.....
```

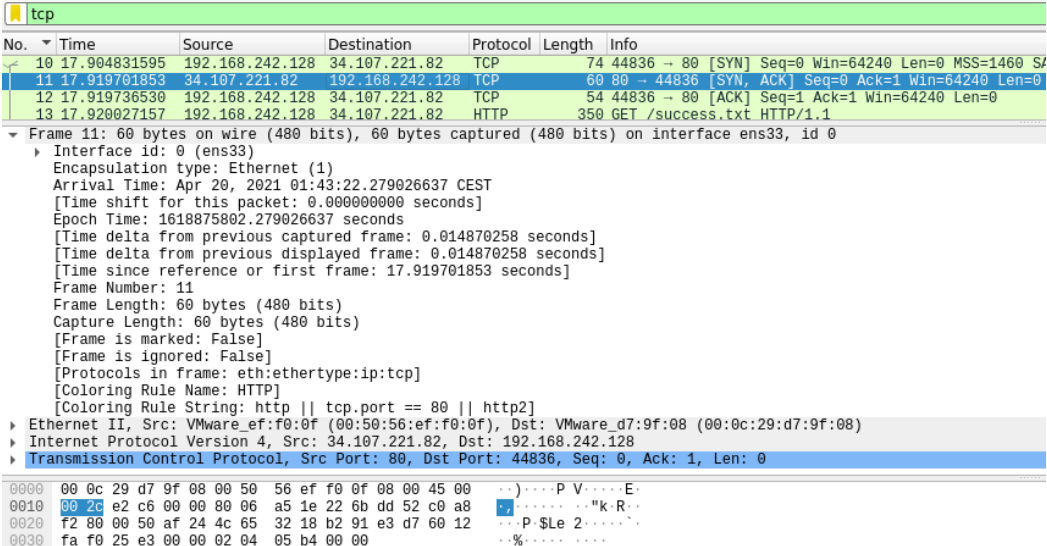


No.	Time	Source	Destination	Protocol	Length	Info
10	17.904831595	192.168.242.128	34.107.221.82	TCP	74	44836 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 S
11	17.919701853	34.107.221.82	192.168.242.128	TCP	60	80 → 44836 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0
12	17.919736530	192.168.242.128	34.107.221.82	TCP	54	44836 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0
13	17.920027157	192.168.242.128	34.107.221.82	HTTP	350	GET /success.txt HTTP/1.1

Frame 10: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface ens33, id 0

- Interface id: 0 (ens33)
- Encapsulation type: Ethernet (1)
- Arrival Time: Apr 20, 2021 01:43:22.264156379 CEST
- [Time shift for this packet: 0.000000000 seconds]
- Epoch Time: 1618875802.264156379 seconds
- [Time delta from previous captured frame: 0.119614689 seconds]
- [Time delta from previous displayed frame: 0.000000000 seconds]
- [Time since reference or first frame: 17.904831595 seconds]
- Frame Number: 10
- Frame Length: 74 bytes (592 bits)
- Capture Length: 74 bytes (592 bits)
- [Frame is marked: False]
- [Frame is ignored: False]
- [Protocols in frame: eth:ethertype:ip:tcp]
- [Coloring Rule Name: HTTP]
- [Coloring Rule String: http || tcp.port == 80 || http2]
- Ethernet II, Src: VMware_ef:f0:0f (00:50:56:ef:f0:0f), Dst: VMware_ef:f0:0f (00:50:56:ef:f0:0f)
- Internet Protocol Version 4, Src: 192.168.242.128, Dst: 34.107.221.82
- Transmission Control Protocol, Src Port: 44836, Dst Port: 80, Seq: 0, Len: 0

0000 00 50 56 ef f0 0f 00 0c 29 d7 9f 08 08 00 45 00 .PV....)....E.
0010 00 3c 65 a7 40 00 40 06 22 2e c0 a8 f2 80 22 6b .<e.@.@. "...."k
0020 dd 52 af 24 00 50 b2 91 e3 d6 00 00 00 00 a0 02 .R.\$P..
0030 fa f0 b3 15 00 00 02 04 05 b4 04 02 08 0a cf a2
0040 78 84 00 00 00 00 01 03 03 07 X.....



No.	Time	Source	Destination	Protocol	Length	Info
10	17.904831595	192.168.242.128	34.107.221.82	TCP	74	44836 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 S
11	17.919701853	34.107.221.82	192.168.242.128	TCP	60	80 → 44836 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0
12	17.919736530	192.168.242.128	34.107.221.82	TCP	54	44836 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0
13	17.920027157	192.168.242.128	34.107.221.82	HTTP	350	GET /success.txt HTTP/1.1

Frame 11: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface ens33, id 0

- Interface id: 0 (ens33)
- Encapsulation type: Ethernet (1)
- Arrival Time: Apr 20, 2021 01:43:22.279026637 CEST
- [Time shift for this packet: 0.000000000 seconds]
- Epoch Time: 1618875802.279026637 seconds
- [Time delta from previous captured frame: 0.014870258 seconds]
- [Time delta from previous displayed frame: 0.014870258 seconds]
- [Time since reference or first frame: 17.919701853 seconds]
- Frame Number: 11
- Frame Length: 60 bytes (480 bits)
- Capture Length: 60 bytes (480 bits)
- [Frame is marked: False]
- [Frame is ignored: False]
- [Protocols in frame: eth:ethertype:ip:tcp]
- [Coloring Rule Name: HTTP]
- [Coloring Rule String: http || tcp.port == 80 || http2]
- Ethernet II, Src: VMware_ef:f0:0f (00:50:56:ef:f0:0f), Dst: VMware_d7:9f:08 (00:0c:29:d7:9f:08)
- Internet Protocol Version 4, Src: 34.107.221.82, Dst: 192.168.242.128
- Transmission Control Protocol, Src Port: 80, Dst Port: 44836, Seq: 0, Ack: 1, Len: 0

0000 00 0c 29 d7 9f 08 00 50 56 ef f0 0f 08 00 45 00 ..)....P V....E.
0010 00 2c e2 c6 00 00 80 06 a5 1e 22 6b dd 52 c0 a8 .,....."k.R..
0020 f2 80 00 50 af 24 4c 65 32 18 b2 91 e3 d7 60 12 ...P.\$Le 2.....
0030 fa f0 25 e3 00 00 02 04 05 b4 00 00 ..%.....

Obrázek 1: Zachytenie dvoch TCP paketov.

6.2 Testovanie UDP paketov – referenčný stroj

```
udp.port==57621
```

No.	Time	Source	Destination	Protocol	Length	Info
12	17.114561854	192.168.242.1	192.168.242.255	UDP	86	57621 → 57621 Len=44

▼ Frame 12: 86 bytes on wire (688 bits), 86 bytes captured (688 bits) on interface ens33, id 0

- Interface id: 0 (ens33)
- Encapsulation type: Ethernet (1)
- Arrival Time: Apr 20, 2021 01:59:02.751262709 CEST
- [Time shift for this packet: 0.000000000 seconds]
- Epoch Time: 1618876742.751262709 seconds
- [Time delta from previous captured frame: 4.111346885 seconds]
- [Time delta from previous displayed frame: 0.000000000 seconds]
- [Time since reference or first frame: 17.114561854 seconds]
- Frame Number: 12
- Frame Length: 86 bytes (688 bits)
- Capture Length: 86 bytes (688 bits)
- [Frame is marked: False]
- [Frame is ignored: False]
- [Protocols in frame: eth:ethertype:ip:udp:data]
- [Coloring Rule Name: UDP]
- [Coloring Rule String: udp]
- Ethernet II, Src: VMware_c0:00:08 (00:50:56:c0:00:08), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
- Internet Protocol Version 4, Src: 192.168.242.1, Dst: 192.168.242.255
- User Datagram Protocol, Src Port: 57621, Dst Port: 57621
- Data (44 bytes)

```
0000  ff ff ff ff ff ff 00 50 56 c0 00 08 08 00 45 00  .....P V.....E
0010  00 48 79 f0 00 00 80 11 5a 62 c0 a8 f2 01 c0 a8  .Hy.....Zb.....
0020  f2 ff e1 15 e1 15 00 34 62 8a 53 70 6f 74 55 64  .....4 b.SpotUd
0030  70 30 9f 10 df 96 b8 2f 9f 69 00 01 00 04 48 95  p0...../ .i....H.
0040  c2 03 39 b0 c5 87 2a 68 86 31 3b 1d e7 05 85 99  ..9...*h .1;.....
0050  d9 87 43 e0 96 2e  ..C...
```

```
student@student-vm:~/ipk-sniffer/run$ sudo ./ipk-sniffer -i ens33 -u -p 57621
2021-04-19T23:59:02.751+02:00 192.168.242.1 : 57621 > 192.168.242.255 : 57621, length 86 bytes
0x0000: ff ff ff ff ff ff 00 50 56 c0 00 08 08 00 45 00  .....P V.....E.
0x0010: 00 48 79 f0 00 00 80 11 5a 62 c0 a8 f2 01 c0 a8  .Hy.....Zb.....
0x0020: f2 ff e1 15 e1 15 00 34 62 8a 53 70 6f 74 55 64  .....4 b.SpotUd
0x0030: 70 30 9f 10 df 96 b8 2f 9f 69 00 01 00 04 48 95  p0...../ .i....H.
0x0040: c2 03 39 b0 c5 87 2a 68 86 31 3b 1d e7 05 85 99  ..9...*h .1;.....
0x0050: d9 87 43 e0 96 2e  ..C...
```

Obrázek 2: Testovanie UDP paketu s portom 57621

6.3 Testovanie ARP paketov – Windows

```
arp
```

No.	Time	Source	Destination	Protocol	Length	Info
163	13.041055	HuaweiTe_77:1f:57	Broadcast	ARP	60	Who has 192.168.100.67? Tell 192.168.100.1

▼ Frame 163: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface \Device\NPF_{E0B7E32E-190E-4752-9A55-A9232E0C338A},

- Interface id: 0 (\Device\NPF_{E0B7E32E-190E-4752-9A55-A9232E0C338A})
- Encapsulation type: Ethernet (1)
- Arrival Time: Apr 19, 2021 22:52:54.317567000 Stredoeurópsky čas (letný)
- [Time shift for this packet: 0.000000000 seconds]
- Epoch Time: 1618865574.317567000 seconds
- [Time delta from previous captured frame: 0.326444000 seconds]
- [Time delta from previous displayed frame: 0.000000000 seconds]
- [Time since reference or first frame: 13.041055000 seconds]
- Frame Number: 163
- Frame Length: 60 bytes (480 bits)
- Capture Length: 60 bytes (480 bits)
- [Frame is marked: False]
- [Frame is ignored: False]
- [Protocols in frame: eth:ethertype:arp]
- [Coloring Rule Name: ARP]
- [Coloring Rule String: arp]
- Ethernet II, Src: HuaweiTe_77:1f:57 (48:fd:8e:77:1f:57), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
- Address Resolution Protocol (request)
- Hardware type: Ethernet (1)
- Protocol type: IPv4 (0x0800)
- Hardware size: 6
- Protocol size: 4
- Opcode: request (1)

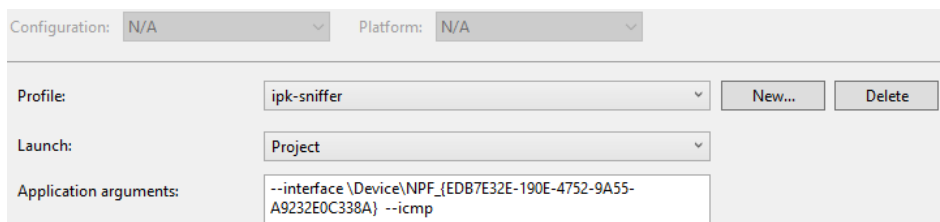
```
0000  ff ff ff ff ff ff 48 fd 8e 77 1f 57 08 06 00 01  .....H. .w.W....
0010  08 00 06 04 00 01 48 fd 8e 77 1f 57 c0 a8 64 01  .....H. .w.W..d.
0020  00 00 00 00 00 00 c0 a8 64 43 6b 6b 6b 6b 6b 6b  .....dCkkkkkkk
0030  6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b  kkkkkkkk kkkk
```

```
Microsoft Visual Studio Debug Console
2021-04-19T20:52:54.317+02:00 192.168.100.1 > 192.168.100.67, length 60 bytes
0x0000: ff ff ff ff ff ff 48 fd 8e 77 1f 57 08 06 00 01  .....H. .w.W....
0x0010: 08 00 06 04 00 01 48 fd 8e 77 1f 57 c0 a8 64 01  .....H. .w.W..d.
0x0020: 00 00 00 00 00 00 c0 a8 64 43 6b 6b 6b 6b 6b 6b  .....dCkkkkkkk
0x0030: 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b 6b  kkkkkkkk kkkk
```

Obrázek 3: Test vykonaný vo Visual Studio 2019

6.4 Testovanie ICMP paketov – Windows

Pre odtestovanie ICMP paketov použijeme príkaz ping.



Obrázek 4: Oba testy boli spustené s týmito vstupnými argumentami

6.4.1 Testovanie ICMPv4 – Windows

Windows PowerShell

Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell <https://aka.ms/pscore6>

PS C:\Users\samue> ping google.com

No.	Time	Source	Destination	Protocol	Length	Info
24	7.0...	192.168.100.152	172.217.23.238	ICMP	74	Echo (ping) request id=0x0001, s
25	7.0...	172.217.23.238	192.168.100.152	ICMP	74	Echo (ping) reply id=0x0001, s
33	8.0...	192.168.100.152	172.217.23.238	ICMP	74	Echo (ping) request id=0x0001, s
35	8.0...	172.217.23.238	192.168.100.152	ICMP	74	Echo (ping) reply id=0x0001, s

▼ Frame 24: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \\Device\\NPF_{EDB7E32E-190E-4752-9A55-A9232E0C338A}

> Interface id: 0 (\\Device\\NPF_{EDB7E32E-190E-4752-9A55-A9232E0C338A})

Encapsulation type: Ethernet (1)

Arrival Time: Apr 19, 2021 22:48:50.053839000 Stredoeurópsky čas (letný)

[Time shift for this packet: 0.000000000 seconds]

Epoch Time: 1618865330.053839000 seconds

[Time delta from previous captured frame: 0.747796000 seconds]

[Time delta from previous displayed frame: 0.000000000 seconds]

[Time since reference or first frame: 7.062054000 seconds]

Frame Number: 24

Frame Length: 74 bytes (592 bits)

Capture Length: 74 bytes (592 bits)

[Frame is marked: False]

[Frame is ignored: False]

[Protocols in frame: eth:ethertype:ip:icmp:data]

[Coloring Rule Name: ICMP]

[Coloring Rule String: icmp || icmpv6]

> Ethernet II, Src: HewlettP_d0:77:e5 (8c:dc:d4:d0:77:e5), Dst: HuaweiTe_77:1f:57 (48:fd:8e:77:1f:57)

▼ Internet Protocol Version 4, Src: 192.168.100.152, Dst: 172.217.23.238

0100 = Version: 4

.... 0101 = Header Length: 20 bytes (5)

Offset	Hex	ASCII
0000	48 fd 8e 77 1f 57 8c dc d4 d0 77 e5 08 00 45 00	H..w.W.. ..w...E.
0010	00 3c b1 77 00 00 80 01 00 00 c0 a8 64 98 ac d9	.<.w....d...
0020	17 ee 08 00 4d 22 00 01 00 39 61 62 63 64 65 66	...M"... .9abcdef
0030	67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76	ghijklmn opqrstuv
0040	77 61 62 63 64 65 66 67 68 69	wabcdefg hi

Microsoft Visual Studio Debug Console

2021-04-19T20:48:50.53+02:00 192.168.100.152 > 172.217.23.238, length 74 bytes

0x0000: 48 fd 8e 77 1f 57 8c dc d4 d0 77 e5 08 00 45 00 H..w.W.. ..w...E.

0x0010: 00 3c b1 77 00 00 80 01 00 00 c0 a8 64 98 ac d9 .<.w....d...

0x0020: 17 ee 08 00 4d 22 00 01 00 39 61 62 63 64 65 66 ...M"... .9abcdef

0x0030: 67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76 ghijklmn opqrstuv

0x0040: 77 61 62 63 64 65 66 67 68 69 wabcdefg hi

6.4.2 Testovanie ICMPv6 – Windows

The image shows two windows from a Windows system. The top window is a PowerShell terminal with a dark blue background. It displays the command `ping 2a00:1450:400a:804::2004` and a list of captured ICMPv6 packets. The bottom window is the Microsoft Visual Studio Debug Console, showing a detailed hex and ASCII dump of the first captured packet (Frame 79).

Windows PowerShell

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\samue> ping 2a00:1450:400a:804::2004
```

icmpv6

No.	Time	Source	Destination	Protocol	Length	Info
79	13....	fe80::7428:143a:b04b:a91c	2a00:1450:400a:804::2004	ICMPv6	94	Echo (ping) request id=0x0000
95	18....	fe80::7428:143a:b04b:a91c	fe80::1	ICMPv6	86	Neighbor Solicitation for fe80::1
96	18....	fe80::7428:143a:b04b:a91c	2a00:1450:400a:804::2004	ICMPv6	94	Echo (ping) request id=0x0000
97	18....	fe80::1	fe80::7428:143a:b04b:a91c	ICMPv6	78	Neighbor Advertisement fe80::1

▼ Frame 79: 94 bytes on wire (752 bits), 94 bytes captured (752 bits) on interface \Device\NPF_{EDB7E32E-190E-4752-9A55-A9232E0C338A}

- > Interface id: 0 (\Device\NPF_{EDB7E32E-190E-4752-9A55-A9232E0C338A})
- Encapsulation type: Ethernet (1)
- Arrival Time: Apr 19, 2021 22:44:42.560175000 Stredoeurópsky čas (letný)
- [Time shift for this packet: 0.000000000 seconds]
- Epoch Time: 1618865082.560175000 seconds
- [Time delta from previous captured frame: 0.047447000 seconds]
- [Time delta from previous displayed frame: 0.000000000 seconds]
- [Time since reference or first frame: 13.857049000 seconds]
- Frame Number: 79
- Frame Length: 94 bytes (752 bits)
- Capture Length: 94 bytes (752 bits)
- [Frame is marked: False]
- [Frame is ignored: False]
- [Protocols in frame: eth:ethertype:ipv6:icmpv6:data]
- [Coloring Rule Name: ICMP]
- [Coloring Rule String: icmp || icmpv6]
- > Ethernet II, Src: HewlettP_d0:77:e5 (8c:dc:d4:d0:77:e5), Dst: HuaweiTe_77:1f:57 (48:fd:8e:77:1f:57)
- > Internet Protocol Version 6, Src: fe80::7428:143a:b04b:a91c, Dst: 2a00:1450:400a:804::2004
- > Internet Control Message Protocol v6

Microsoft Visual Studio Debug Console

```
2021-04-19T20:44:42.560+02:00 fe80::7428:143a:b04b:a91c > 2a00:1450:400a:804::2004, length 94 bytes
0x0000: 48 fd 8e 77 1f 57 8c dc d4 d0 77 e5 86 dd 60 00 H..w.W.. ..w...`
0x0010: 00 00 00 28 3a 80 fe 80 00 00 00 00 00 00 74 28 ...(:... ..t(
0x0020: 14 3a b0 4b a9 1c 2a 00 14 50 40 0a 08 04 00 00 ...K...*. .P@....
0x0030: 00 00 00 00 20 04 80 00 4d e7 00 01 00 63 61 62 .... ..M....cab
0x0040: 63 64 65 66 67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 cdefghij klmnopqr
0x0050: 73 74 75 76 77 61 62 63 64 65 66 67 68 69 stuvwabc defghi
```

Obrázek 5: Zachytenie zdrojovej a cieľovej IPv6 adresy

7 Záver

Okrem týchto vzorových testov prebehlo mnoho ďalších počas celej doby implementácie ale v dokumentácii som sa snažil zamerať na predvedenie všetkých prepínačov. Rovnako som sa snažil v testovaní ukázať multiplatformné využitie môjho riešenia. Implementácia ma bavila, aj keď bolo potrebné si predom naštudovať množstvo teoretických informácií. Na záver by som chcel ešte rád prezentovať zachytenie UDP IPv6 paketu.

```
3... 9.442713 fe80::7428:143a:b0... ff02::c UDP 718 52170 → 3702 Len=656
<
▼ Frame 346: 718 bytes on wire (5744 bits), 718 bytes captured (5744 bits) on interface \Device\
  > Interface id: 0 (\Device\NPF_{EDB7E32E-190E-4752-9A55-A9232E0C338A})
    Encapsulation type: Ethernet (1)
    Arrival Time: Apr 20, 2021 17:40:46.124638000 Stredoeurópsky čas (letný)
    [Time shift for this packet: 0.000000000 seconds]
    Epoch Time: 1618933246.124638000 seconds
    [Time delta from previous captured frame: 0.150294000 seconds]
    [Time delta from previous displayed frame: 1.644687000 seconds]
    [Time since reference or first frame: 9.442713000 seconds]
    Frame Number: 346
    Frame Length: 718 bytes (5744 bits)
    Capture Length: 718 bytes (5744 bits)
    [Frame is marked: False]
    [Frame is ignored: False]
    [Protocols in frame: eth:ethertype:ipv6:udp:data]
    [Coloring Rule Name: UDP]
    [Coloring Rule String: udp]
  > Ethernet II, Src: HewlettP_d0:77:e5 (8c:dc:d4:d0:77:e5), Dst: IPv6mcast_0c (33:33:00:00:00:0c)
  > Internet Protocol Version 6, Src: fe80::7428:143a:b04b:a91c, Dst: ff02::c
  > User Datagram Protocol, Src Port: 52170, Dst Port: 3702
  > Data (656 bytes)
0000 33 33 00 00 00 0c 8c dc d4 d0 77 e5 86 dd 60 03 33.....w...
0010 dc a5 02 98 11 01 fe 80 00 00 00 00 00 00 74 28 .....t(
0020 14 3a b0 4b a9 1c ff 02 00 00 00 00 00 00 00 00 ...K.....
0030 00 00 00 00 00 0c cb ca 0e 76 02 98 08 30 3c 3f .....v...0<?
0040 78 6d 6c 20 76 65 72 73 69 6f 6e 3d 22 31 2e 30 xml vers ion="1.0
0050 22 20 65 6e 63 6f 64 69 6e 67 3d 22 75 74 66 2d " encodi ng="utf-
0060 38 22 3f 3e 3c 73 6f 61 70 3a 45 6e 76 65 6c 6f 8"?><soa p:Envelo
0070 70 65 20 78 6d 6c 6e 73 3a 73 6f 61 70 3d 22 68 pe xmlns :soap="h
0080 74 74 70 3a 2f 2f 77 77 77 2e 77 33 2e 6f 72 67 ttp://ww w.w3.org
0090 2f 32 30 30 33 2f 30 35 2f 73 6f 61 70 2d 65 6e /2003/05 /soap-en
00a0 76 65 6c 6f 70 65 22 20 78 6d 6c 6e 73 3a 77 73 velope" xmlns:ws

Microsoft Visual Studio Debug Console
2021-04-20T15:40:46.124+02:00 fe80::7428:143a:b04b:a91c : 52170 > ff02::c : 3702, length 718 bytes
0x0000: 33 33 00 00 00 0c 8c dc d4 d0 77 e5 86 dd 60 03 33.....w...
0x0010: dc a5 02 98 11 01 fe 80 00 00 00 00 00 00 74 28 .....t(
0x0020: 14 3a b0 4b a9 1c ff 02 00 00 00 00 00 00 00 00 ...K.....
0x0030: 00 00 00 00 00 0c cb ca 0e 76 02 98 08 30 3c 3f .....v...0<?
0x0040: 78 6d 6c 20 76 65 72 73 69 6f 6e 3d 22 31 2e 30 xml vers ion="1.0
0x0050: 22 20 65 6e 63 6f 64 69 6e 67 3d 22 75 74 66 2d " encodi ng="utf-
0x0060: 38 22 3f 3e 3c 73 6f 61 70 3a 45 6e 76 65 6c 6f 8"?><soa p:Envelo
0x0070: 70 65 20 78 6d 6c 6e 73 3a 73 6f 61 70 3d 22 68 pe xmlns :soap="h
0x0080: 74 74 70 3a 2f 2f 77 77 77 2e 77 33 2e 6f 72 67 ttp://ww w.w3.org
0x0090: 2f 32 30 30 33 2f 30 35 2f 73 6f 61 70 2d 65 6e /2003/05 /soap-en
0x00a0: 76 65 6c 6f 70 65 22 20 78 6d 6c 6e 73 3a 77 73 velope" xmlns:ws
```

Obrázek 6: Ukážka UDP paketu pre IPv6, vzhľadom k dĺžke som vybral pre prezentáciu iba časť.

8 Odkazy na referencie

Reference

- [1] C# (CSharp) SharpPcap Namespace. [online], Dostupné z: <https://csharpdoc.hotexamples.com/namespace/SharpPcap>, navštívené 2021-04-16.
- [2] LEIBOWITZ, M.: Getting Started with System.CommandLine. [online], 2020, Dostupné z: <https://dotnetdevaddict.co.za/2020/09/25/getting-started-with-system-commandline/>, navštívené 2021-04-15.
- [3] MORGAN, C.: Packet.Net. [online], 2021, Dostupné z: <https://github.com/chmorgan/packetnet>, navštívené 2021-04-16.
- [4] MULLINS, M.: Exploring the anatomy of a data packet. [online], 2001, Dostupné z: <https://www.techrepublic.com/article/exploring-the-anatomy-of-a-data-packet/>, navštívené 2021-04-16.
- [5] Wikipedia: Address Resolution Protocol. [online], 2021, Dostupné z: https://cs.wikipedia.org/wiki/Address_Resolution_Protocol, navštívené 2021-04-17.
- [6] Wikipedia: Ethernet. [online], 2021, Dostupné z: <https://cs.wikipedia.org/wiki/Ethernet>, navštívené 2021-04-16.
- [7] ZYDYK, M.: Address Resolution Protocol (ARP). [online], Dostupné z: <https://searchnetworking.techtarget.com/definition/Address-Resolution-Protocol-ARP>, navštívené 2021-04-18.
- [8] ŠIPKOVSKÝ, M.: Najpoužívanéjšie TCP a UDP porty – Well known ports. [online], 2018, Dostupné z: <https://netvel.sk/well-known-ports/>, navštívené 2021-04-17.