

Warstwa łącza danych

Sprawozdanie nr 3 z przedmiotu Sieci Komputerowe

Maciej Kaszkowiak, 151856, zadania wykonane 1 kwietnia 2023

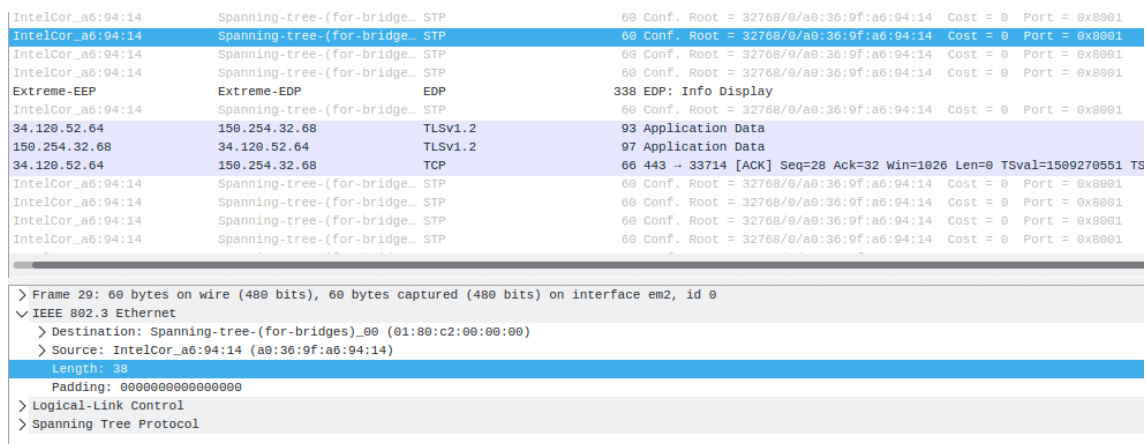
Spis treści

1	Zadanie 1	2
1.1	Przy pomocy programu wireshark przeanalizuj ruch w sieci pod kątem występowania ramek Ethernet 802.3 i Ethernet v2.	2
1.2	W przypadku jakich pakietów daje się zaobserwować ramki Ethernet 802.3?	2
2	Zadanie 2	2
2.1	Podłącz swój komputer (poprzez port p4p1) do koncentratora (na zapleczu). . . .	2
2.2	Skonfiguruj interfejs p4p1, tak by wszystkie komputery w rzędzie działały w jednej sieci (unikalne sieci między rzędami).	2
2.3	Przy pomocy programu wireshark przeanalizuj ruch na interfejsie p4p1. Czy daje się zaobserwować pakiety z innych sieci (w szczególności komunikaty protokołu telnet)?	3
3	Zadanie 3	3
3.1	W jednej chwili (wraz z koleżankami/kolegami) dokonaj pomiaru prędkości transmisji z sąsiednim komputerem z tego samego rzędu.	3
3.2	Wraz z innymi studentami zbadaj sumaryczną przepustowość koncentratora.	4
4	Zadanie 4	4
4.1	Przepnij (na zapleczu) swój komputer do wspólnego przełącznika i dokonaj ponownie pomiaru prędkości (jak w poprzednim zadaniu).	4
4.2	Skąd wynikają różnice w osiągniętych prędkościach?	5
4.3	Czy w programie Wireshark można zobaczyć pakiety z innych sieci? Jeśli tak to jakie?	5
5	Zadanie 5	5
5.1	Wraz z koleżankami/kolegami przepnij połowę komputerów do drugiego przełącznika, tak by komputery w parach były podłączone do różnych przełączników. . . .	5
5.2	Przełączniki połącz ze sobą kablem (krosowanym).	5
5.3	Dokonaj ponownie pomiaru prędkości (jak w poprzednim zadaniu).	5
5.4	Skąd wynikają różnice w osiągniętych prędkościach?	6
5.5	Czy można łatwo zwiększyć przepustowość?	6
6	Zadanie 6	6
6.1	Dokonaj pomiaru prędkości transmisji do adresu 127.0.0.1.	6
6.2	Czy do tej transmisji angażowana jest karta sieciowa?	6
7	Zadanie 7	6
7.1	Podepnij swój komputer na porcie p4p1 do przełącznika, tak by jeden przełącznik przypadał na jeden rząd w laboratorium. Upewnij się, że działa połączenie z innymi komputerami w rzędzie.	6
7.2	Zepnij dwoma kablami przełącznik, do którego jest podłączony Twój komputer z przełącznikiem odpowiadającym sąsiedniemu rzędowi.	7
7.3	Odpowiednio skonfiguruj interfejs p4p1 by móc skomunikować się z siecią z sąsiedniego rzędu lub by dało się zaobserwować jakiegokolwiek ruch w tamtej sieci.	7
7.4	Podglądaj ruch na interfejsie p4p1 przy pomocy programu wireshark (najlepiej robić to tylko na jednym lub dwóch wybranych komputerach w rzędzie).	7
7.5	Jeden z komputerów w rzędzie podepnij do wejścia konsolowego przełącznika i uruchom (jako root) polecenie <code>picocom /dev/ttyS0</code> . Zmień ustawienia/wykonaj polecenia na przełączniku:	8
7.6	Jeśli nie działałoby się nic spektakularnego, wykonaj komendę <code>ping</code> na adres z sieci zdefiniowanej dla rzędu, ale który jest nieprzydzielony żadnemu komputerowi w laboratorium.	8

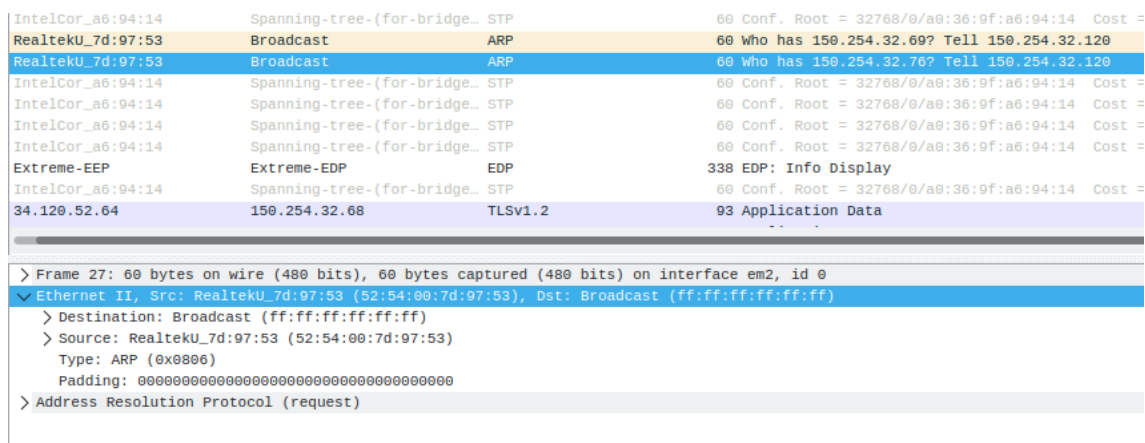
1 Zadanie 1

1.1 Przy pomocy programu wireshark przeanalizuj ruch w sieci pod kątem występowania ramek Ethernet 802.3 i Ethernet v2.

W sieci występują ramki zarówno typu Ethernet 802.3 jak i Ethernet v2.



Rysunek 1: Ramka STP typu Ethernet 802.3



Rysunek 2: Ramka ARP typu Ethernet v2

1.2 W przypadku jakich pakietów daje się zaobserwować ramki Ethernet 802.3?

Możemy zaobserwować ramki Ethernet 802.3 między innymi w przypadku pakietów z protokołami EDP, DTP, STP (Spanning Tree Protocol).

2 Zadanie 2

2.1 Podłącz swój komputer (poprzez port p4p1) do koncentratora (na zapleczu).

Podłączyłem komputer poprzez port p4p1 do koncentratora na zapleczu.

2.2 Skonfiguruj interfejs p4p1, tak by wszystkie komputery w rzędzie działały w jednej sieci (unikalne sieci między rzędami).

Wykonałem następującą komendę:

```
lab-sec-3:/homex/student # ip addr add 192.168.1.3/24 dev p4p1
```

2.3 Przy pomocy programu wireshark przeanalizuj ruch na interfejsie p4p1. Czy daje się zaobserwować pakiety z innych sieci (w szczególności komunikaty protokołu telnet)?

Tak, mogłem zaobserwować pakiety z innych sieci, w szczególności z protokołu telnet.

Source	Destination	Protocol	Length	Info
192.168.5.14	192.168.5.15	TELNET	67	Telnet Data ...
192.168.5.15	192.168.5.14	TCP	66	23 → 50756 [ACK] Seq=152 Ack=207 Win=65024 Len=0 TSval=23:
192.168.5.14	192.168.5.15	TELNET	67	Telnet Data ...
192.168.5.15	192.168.5.14	TCP	66	23 → 50756 [ACK] Seq=152 Ack=208 Win=65024 Len=0 TSval=23:
192.168.5.14	192.168.5.15	TELNET	67	Telnet Data ...
192.168.5.15	192.168.5.14	TCP	66	23 → 50756 [ACK] Seq=152 Ack=209 Win=65024 Len=0 TSval=23:
192.168.5.15	192.168.5.14	ICMP	98	Echo (ping) request id=0x0002, seq=24/6144, ttl=64 (repl:
192.168.5.14	192.168.5.15	ICMP	98	Echo (ping) reply id=0x0002, seq=24/6144, ttl=64 (requ
192.168.5.14	192.168.5.15	TELNET	68	Telnet Data ...
192.168.5.15	192.168.5.14	TCP	66	23 → 50756 [ACK] Seq=152 Ack=211 Win=65024 Len=0 TSval=23:
192.168.5.15	192.168.5.14	TELNET	68	Telnet Data ...
192.168.5.14	192.168.5.15	TCP	66	50756 → 23 [ACK] Seq=211 Ack=154 Win=64256 Len=0 TSval=39:
192.168.5.15	192.168.5.14	TELNET	129	Telnet Data ...
192.168.5.14	192.168.5.15	TCP	66	50756 → 23 [ACK] Seq=211 Ack=217 Win=64256 Len=0 TSval=39:
192.168.5.15	192.168.5.14	TELNET	124	Telnet Data ...
192.168.5.14	192.168.5.15	TCP	66	50756 → 23 [ACK] Seq=211 Ack=275 Win=64256 Len=0 TSval=39:
192.168.5.15	192.168.5.14	TELNET	88	Telnet Data ...
192.168.5.14	192.168.5.15	TCP	66	50756 → 23 [ACK] Seq=211 Ack=297 Win=64256 Len=0 TSval=39:
192.168.5.15	192.168.5.14	TELNET	103	Telnet Data ...
192.168.5.14	192.168.5.15	TCP	66	50756 → 23 [ACK] Seq=211 Ack=334 Win=64256 Len=0 TSval=39:
Broadcom_da:4d:48	Broadcom_7f:15:18	ARP	60	Who has 192.168.4.13? Tell 192.168.4.12
Broadcom_7f:15:18	Broadcom_da:4d:48	ARP	60	192.168.4.13 is at 00:10:18:7f:15:18
192.168.2.8	192.168.2.7	ICMP	98	Echo (ping) request id=0x0002, seq=92/23552, ttl=64 (rep.
192.168.2.7	192.168.2.8	ICMP	98	Echo (ping) reply id=0x0002, seq=92/23552, ttl=64 (req
192.168.5.15	192.168.5.14	ICMP	98	Echo (ping) request id=0x0002, seq=25/6400, ttl=64 (repl:
192.168.5.14	192.168.5.15	ICMP	98	Echo (ping) reply id=0x0002, seq=25/6400, ttl=64 (requ
192.168.2.8	192.168.2.7	ICMP	98	Echo (ping) request id=0x0002, seq=93/23808, ttl=64 (rep.

Rysunek 3: Pakiety protokołu Telnet.

Musiałem również usunąć poprzednią konfigurację, a także wpiąć ponownie kabel do przełącznika (odpięcie wywołało stan DOWN).

```
lab-sec-3:/homex/student # ip addr del 192.168.1.3/16 dev p4p1
```

Po wpięciu kabla mogłem nawiązać ponownie transmisję:

```
lab-sec-3:/homex/student # ping 192.168.1.2
PING 192.168.1.2 (192.168.1.2) 56(84) bytes of data.
64 bytes from 192.168.1.2: icmp_seq=1 ttl=64 time=0.945 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=64 time=0.709 ms
64 bytes from 192.168.1.2: icmp_seq=3 ttl=64 time=0.713 ms
```

3 Zadanie 3

3.1 W jednej chwili (wraz z koleżankami/kolegami) dokonaj pomiaru prędkości transmisji z sąsiednim komputerem z tego samego rzędu.

Trzykrotnie uruchomiłem klient pomiaru transmisji w odstępach czasu. Pomiary prędkości różniły się w przedziale od 3 do 8 Mb/s.

```
lab-sec-3:/homex/student # netperf -H 192.168.1.2
MIGRATED TCP STREAM TEST from 0.0.0.0 (0.0.0.0) port 0 AF_INET to 192.168.1.2 () port 0 AF_INET :
Recv  Send  Send
Socket Socket Message Elapsed
Size  Size  Size  Time   Throughput
bytes bytes bytes secs.  10^6bits/sec

131072 16384 16384 12.23   8.23
```

```
lab-sec-3:/homex/student # netperf -H 192.168.1.2
MIGRATED TCP STREAM TEST from 0.0.0.0 (0.0.0.0) port 0 AF_INET to 192.168.1.2 () port 0 AF_INET :
Recv  Send    Send
Socket Socket  Message Elapsed
Size  Size    Size    Time    Throughput
bytes bytes  bytes   secs.   10^6bits/sec

131072 16384 16384    12.98     6.35
lab-sec-3:/homex/student # netperf -H 192.168.1.2
MIGRATED TCP STREAM TEST from 0.0.0.0 (0.0.0.0) port 0 AF_INET to 192.168.1.2 () port 0 AF_INET :
Recv  Send    Send
Socket Socket  Message Elapsed
Size  Size    Size    Time    Throughput
bytes bytes  bytes   secs.   10^6bits/sec

131072 16384 16384    10.88     3.38
```

Wraz z rosnącą liczbą ludzi testujących jednocześnie przepustowość, można zauważyć że przepustowość w obrębie rzędu sukcesywnie malała.

3.2 Wraz z innymi studentami zbadaj sumaryczną przepustowość koncentratora.

Wraz z innymi studentami zbadaliśmy sumaryczną przepustowość koncentratora. Do 8 kwietnia włącznie swoje odpowiedzi przesłały 4 osoby (6 Mb/s, 3.43 Mb/s, 5.56 Mb/s, 6.95 Mb/s). Suma przesłanych przepustowości wynosi 21.94 Mb/s - uważam natomiast, że nie stanowi to sumarycznej przepustowości koncentratora. Pomiary nie zostały wykonane w jednym momencie czasu, przez co chwilowa domena kolizyjna mogła być znacznie mniejsza. Na korzyść wymienionej tezy pragnę zauważyć, że pomiary wykonane przez mój rząd różniły się w zakresie od 3.38 do 8.23 Mb/s, w zależności od momentu uruchomienia testu.

4 Zadanie 4

4.1 Przepnij (na zapleczu) swój komputer do wspólnego przełącznika i dokonaj ponownie pomiaru prędkości (jak w poprzednim zadaniu).

Możemy zaobserwować, że prędkość znacząco wzrosła do poziomu 941 Mb/s. (około 100-150 razy). Przede wszystkim: prędkość sieci jest stabilna oraz powtarzana pomiędzy testami.

```
lab-sec-3:/homex/student # netperf -H 192.168.1.2
MIGRATED TCP STREAM TEST from 0.0.0.0 (0.0.0.0) port 0 AF_INET to 192.168.1.2 () port 0 AF_INET :
Recv  Send    Send
Socket Socket  Message Elapsed
Size  Size    Size    Time    Throughput
bytes bytes  bytes   secs.   10^6bits/sec

131072 16384 16384    10.03    941.36
lab-sec-3:/homex/student # netperf -H 192.168.1.2
MIGRATED TCP STREAM TEST from 0.0.0.0 (0.0.0.0) port 0 AF_INET to 192.168.1.2 () port 0 AF_INET :
Recv  Send    Send
Socket Socket  Message Elapsed
Size  Size    Size    Time    Throughput
bytes bytes  bytes   secs.   10^6bits/sec

131072 16384 16384    10.03    941.36
lab-sec-3:/homex/student # netperf -H 192.168.1.2
MIGRATED TCP STREAM TEST from 0.0.0.0 (0.0.0.0) port 0 AF_INET to 192.168.1.2 () port 0 AF_INET :
Recv  Send    Send
Socket Socket  Message Elapsed
```

Size bytes	Size bytes	Size bytes	Time secs.	Throughput 10 ⁶ bits/sec
131072	16384	16384	10.03	941.36

4.2 Skąd wynikają różnice w osiąganych prędkościach?

Różnice w osiąganych prędkościach pomiędzy switchem a hubem spowodowane są nadmiernymi kolizjami w sieciach opartych o koncentrator.

4.3 Czy w programie Wireshark można zobaczyć pakiety z innych sieci? Jeśli tak to jakie?

W programie Wireshark można zobaczyć pakiety w innych sieci, jednak są to pakiety tylko wybranych protokołów - zauważyłem m.in. protokoły SSDP.

5 Zadanie 5

5.1 Wraz z koleżankami/kolegami przepnij połowę komputerów do drugiego przełącznika, tak by komputery w parach były podłączone do różnych przełączników.

Mój komputer z numerem nieparzystym pozostał wpięty do pierwszego przełącznika.

5.2 Przełączniki połącz ze sobą kablem (krosowanym).

Dyżurny połączył przełączniki kablem krosowanym.

5.3 Dokonaj ponownie pomiaru prędkości (jak w poprzednim zadaniu).

Pomiar prędkości na początku był niewykonalny:

```
lab-sec-3:/homex/student # ping 192.168.1.2
PING 192.168.1.2 (192.168.1.2) 56(84) bytes of data.
64 bytes from 192.168.1.2: icmp_seq=79 ttl=64 time=81.4 ms
^C
--- 192.168.1.2 ping statistics ---
147 packets transmitted, 1 received, 99.3197% packet loss, time 149466ms
rtt min/avg/max/mdev = 81.398/81.398/81.398/0.000 ms
```

Po zmianach prowadzącego w połączeniach sieć zaczęła działać. Powód problemów z siecią: ktoś wpiął dwa końce jednego kabla do przełącznika.

```
lab-sec-3:/homex/student # ping 192.168.1.2
PING 192.168.1.2 (192.168.1.2) 56(84) bytes of data.
64 bytes from 192.168.1.2: icmp_seq=20 ttl=64 time=7.76 ms
64 bytes from 192.168.1.2: icmp_seq=39 ttl=64 time=36.2 ms
64 bytes from 192.168.1.2: icmp_seq=39 ttl=64 time=36.5 ms (DUP!)
From 192.168.1.3 icmp_seq=74 Destination Host Unreachable
From 192.168.1.3 icmp_seq=75 Destination Host Unreachable
From 192.168.1.3 icmp_seq=76 Destination Host Unreachable
[...]
From 192.168.1.3 icmp_seq=95 Destination Host Unreachable
From 192.168.1.3 icmp_seq=96 Destination Host Unreachable
From 192.168.1.3 icmp_seq=97 Destination Host Unreachable
64 bytes from 192.168.1.2: icmp_seq=98 ttl=64 time=1025 ms
64 bytes from 192.168.1.2: icmp_seq=99 ttl=64 time=1.31 ms
64 bytes from 192.168.1.2: icmp_seq=100 ttl=64 time=0.698 ms
64 bytes from 192.168.1.2: icmp_seq=101 ttl=64 time=0.706 ms
[...]
```

```

64 bytes from 192.168.1.2: icmp_seq=131 ttl=64 time=0.713 ms
64 bytes from 192.168.1.2: icmp_seq=132 ttl=64 time=0.711 ms
64 bytes from 192.168.1.2: icmp_seq=133 ttl=64 time=0.340 ms
^C
--- 192.168.1.2 ping statistics ---
133 packets transmitted, 33 received, +1 duplicates, +24 errors, 75.188% packet loss, time 134969ms
rtt min/avg/max/mdev = 0.330/33.248/1025.326/172.902 ms, pipe 4

```

Przeprowadziliśmy test prędkości:

```

lab-sec-3:/homex/student # netperf -H 192.168.1.2
MIGRATED TCP STREAM TEST from 0.0.0.0 (0.0.0.0) port 0 AF_INET to 192.168.1.2 () port 0 AF_INET : de
Recv  Send  Send
Socket Socket Message Elapsed
Size  Size  Size  Time  Throughput
bytes bytes bytes secs.  10^6bits/sec

131072 16384 16384 10.03 941.36

```

5.4 Skąd wynikają różnice w osiągniętych prędkościach?

Prędkość mogłaby wzrosnąć ze względu na mniejszą domenę kolizyjną, jednak w teście pozostała ta sama.

5.5 Czy można łatwo zwiększyć przepustowość?

Tak, rozdzielając sieć na mniejsze podsieci.

6 Zadanie 6

6.1 Dokonaj pomiaru prędkości transmisji do adresu 127.0.0.1.

```

lab-sec-3:/homex/student # netperf -H 127.0.0.1
MIGRATED TCP STREAM TEST from 0.0.0.0 (0.0.0.0) port 0 AF_INET to 127.0.0.1 () port 0 AF_INET : de
Recv  Send  Send
Socket Socket Message Elapsed
Size  Size  Size  Time  Throughput
bytes bytes bytes secs.  10^6bits/sec

131072 16384 16384 10.00 90918.49

```

6.2 Czy do tej transmisji angażowana jest karta sieciowa?

Nie, pętlę zwrotną można wykorzystywać nawet bez karty sieciowej.

7 Zadanie 7

7.1 Podepnij swój komputer na porcie p4p1 do przełącznika, tak by jeden przełącznik przypadał na jeden rząd w laboratorium. Upewnij się, że działa połączenie z innymi komputerami w rzędzie.

Skonfigurowałem połączenie na porcie p4p1. Połączenie z innymi komputerami działa:

```

lab-sec-3:/homex/student # ping 192.168.1.2
PING 192.168.1.2 (192.168.1.2) 56(84) bytes of data.
64 bytes from 192.168.1.2: icmp_seq=1 ttl=64 time=0.178 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=64 time=0.206 ms
64 bytes from 192.168.1.2: icmp_seq=3 ttl=64 time=0.514 ms
64 bytes from 192.168.1.2: icmp_seq=4 ttl=64 time=0.254 ms
^C

```

```
--- 192.168.1.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3059ms
rtt min/avg/max/mdev = 0.178/0.288/0.514/0.133 ms
```

7.2 Zepnij dwoma kablami przełącznik, do którego jest podłączony Twój komputer z przełącznikiem odpowiadającym sąsiadnemu rzędowni.

Spieiliśmy sąsiadujące przełączniki za pomocą kabli krosowych.

7.3 Odpowiednio skonfiguruj interfejs p4p1 by móc skomunikować się z siecią z sąsiedniego rzędu lub by dało się zaobserwować jakikolwiek ruch w tamtej sieci.

Skonfigurowałem interfejs p4p1 za pomocą następujących komend:

```
lab-sec-3:/homex/student # ip addr del 192.168.1.3/24 dev p4p1
lab-sec-3:/homex/student # ip addr add 192.168.1.3/16 dev p4p1
```

Komunikacja nadal działa, zarówno w obrębie rzędu jak i całej sali:

```
lab-sec-3:/homex/student # ping 192.168.1.2
PING 192.168.1.2 (192.168.1.2) 56(84) bytes of data.
64 bytes from 192.168.1.2: icmp_seq=1 ttl=64 time=0.367 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=64 time=0.312 ms
64 bytes from 192.168.1.2: icmp_seq=3 ttl=64 time=0.457 ms
64 bytes from 192.168.1.2: icmp_seq=4 ttl=64 time=0.696 ms
64 bytes from 192.168.1.2: icmp_seq=5 ttl=64 time=0.699 ms
^C
--- 192.168.1.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4082ms
rtt min/avg/max/mdev = 0.312/0.506/0.699/0.162 ms
lab-sec-3:/homex/student # ping 192.168.2.5
PING 192.168.2.5 (192.168.2.5) 56(84) bytes of data.
64 bytes from 192.168.2.5: icmp_seq=1 ttl=64 time=1.07 ms
64 bytes from 192.168.2.5: icmp_seq=2 ttl=64 time=0.693 ms
64 bytes from 192.168.2.5: icmp_seq=3 ttl=64 time=0.693 ms
64 bytes from 192.168.2.5: icmp_seq=4 ttl=64 time=0.247 ms
^C
--- 192.168.2.5 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3041ms
rtt min/avg/max/mdev = 0.247/0.675/1.068/0.290 ms
```

7.4 Podglądaj ruch na interfejsie p4p1 przy pomocy programu wireshark (najlepiej robić to tylko na jednym lub dwóch wybranych komputerach w rzędzie).

Możemy zaobserwować ruch w obrębie całej sali:

Source	Destination	Protocol	Length	Info
Broadcast_7f:15:18	Broadcast	ARP	60	who has 192.168.5.15? Tell 192.168.4.13
Cisco_35:40:05	Cisco_35:40:05	LOOP	60	Reply
Cisco_35:40:05	Spanning-tree-(for-bridge_	STP	60	Conf. Root = 32768/1/00:1b:54:a3:03:80 Cost = 4 Port = 0x8017
Broadcast_ae:13:04	Broadcast	ARP	60	who has 192.168.5.14? Tell 192.168.3.10
Cisco_35:40:05	Spanning-tree-(for-bridge_	STP	60	Conf. Root = 32768/1/00:1b:54:a3:03:80 Cost = 4 Port = 0x8017
Cisco_35:40:05	Spanning-tree-(for-bridge_	STP	60	Conf. Root = 32768/1/00:1b:54:a3:03:80 Cost = 4 Port = 0x8017
Cisco_35:40:05	Spanning-tree-(for-bridge_	STP	60	Conf. Root = 32768/1/00:1b:54:a3:03:80 Cost = 4 Port = 0x8017
Cisco_35:40:05	Spanning-tree-(for-bridge_	STP	60	Conf. Root = 32768/1/00:1b:54:a3:03:80 Cost = 4 Port = 0x8017
192.168.5.15	239.255.255.250	SSDP	214	M-SEARCH * HTTP/1.1
192.168.5.15	239.255.255.250	SSDP	214	M-SEARCH * HTTP/1.1
Cisco_35:40:05	Cisco_35:40:05	LOOP	60	Reply
Broadcast_b3:4f:dc	Broadcast	ARP	60	who has 192.168.1.3? Tell 192.168.2.7
Broadcast_b4:e3:7c	Broadcast	ARP	42	192.168.1.3 is at 00:10:18:b4:e3:7c
192.168.2.7	192.168.1.3	ICMP	98	Echo (ping) request id=0x0024, seq=1/256, ttl=64 (reply in 28)
192.168.1.3	192.168.2.7	ICMP	98	Echo (ping) reply id=0x0024, seq=1/256, ttl=64 (request in 27)
Cisco_35:40:05	Spanning-tree-(for-bridge_	STP	60	Conf. Root = 32768/1/00:1b:54:a3:03:80 Cost = 4 Port = 0x8017
192.168.5.15	239.255.255.250	SSDP	214	M-SEARCH * HTTP/1.1
Broadcast_da:4d:48	Broadcast	ARP	60	who has 192.168.3.11? Tell 192.168.4.12
192.168.2.7	192.168.1.3	ICMP	98	Echo (ping) request id=0x0024, seq=2/512, ttl=64 (reply in 33)
192.168.1.3	192.168.2.7	ICMP	98	Echo (ping) reply id=0x0024, seq=2/512, ttl=64 (request in 32)
192.168.5.15	239.255.255.250	SSDP	214	M-SEARCH * HTTP/1.1

Rysunek 4: Ruch na interfejsie p4p1.

7.5 Jeden z komputerów w rzędzie podejmuje do wejścia konsolowego przełącznika i uruchomi (jako root) polecenie picocom /dev/ttyS0. Zmień ustawienia/wykonaj polecenia na przełączniku:

- terminate auto install: yes
- enter auto configuration: no
- enable
- configure terminal
- no spanning-tree vlan 1

Zgodnie z poleceniem prowadzącego, zadanie zostało niewykonane przez brak czasu.

7.6 Jeśli nie działałoby się nic spektakularnego, wykonaj komendę ping na adres z sieci zdefiniowanej dla rzędu, ale który jest nieprzydzielony żadnemu komputerowi w laboratorium.

Zgodnie z poleceniem prowadzącego, zadanie zostało niewykonane przez brak czasu.