

Podstawowe narzędzia sieciowe w Linuksie

Sprawozdanie nr 2 z przedmiotu Sieci Komputerowe

Maciej Kaszkowiak, 151856, zadania wykonane 1 kwietnia 2023

Spis treści

1	Zadanie 1	2
1.1	Wspólnie z koleżankami/kolegami z rzędu ustal adres sieci, który pozwoli na zaadresowanie wszystkich komputerów w rzędzie. Sieci są unikalne między rzędami.	2
1.2	Przypisz jeden z adresów do interfejsu em1 (każdy komputer w rzędzie ma mieć inny adres).	2
1.3	Przeanalizuj ruch na interfejsie em1 przy pomocy programu wireshark.	2
1.4	Zbadaj przy pomocy polecenia ping połączenie z innymi komputerami w rzędzie. .	2
1.5	Zbadaj przy pomocy polecenia ping połączenie komputerami w innych rzędach. . .	2
1.6	Parami połącz się z innym komputerem z rzędu przy pomocy komendy telnet i zbadaj treść wymienianych pakietów.	3
2	Zadanie 2	4
2.1	Zmierz osiąganą maksymalną prędkość transmisji przy użyciu usługi speedtest.net.	4
2.2	Zmień parametry em1 ograniczając prędkość transmisji i zmieniając parametry duplexu.	5
2.3	Ponownie zmierz maksymalną prędkość transmisji. Wróć do pierwotnych ustawień. Przy pomocy odpowiedniego parametru komendy ethtool zidentyfikuj interfejsy p4p1 oraz p4p2.	5
3	Zadanie 3	6
3.1	Usuń nadane wcześniej adresy IP na em1 (mają pozostać tylko standardowe adresy, tj. uzyskiwane przez DHCP).	6
3.2	Podłącz swój komputer poprzez port p4p1 do przełącznika na zapleczu (wszystkie komputery są podłączane do tego samego przełącznika).	6
3.3	Skonfiguruj adres IP na porcie p4p1, tak by wszystkie komputery w laboratorium mogły komunikować się między sobą poprzez nowo utworzoną sieć	6

1 Zadanie 1

Poniższe polecenia wykonywane są na interfejsie br0.

1.1 Wspólnie z koleżankami/kolegami z rzędu ustal adres sieci, który pozwoli na zaadresowanie wszystkich komputerów w rzędzie. Sieci są unikalne między rzędami.

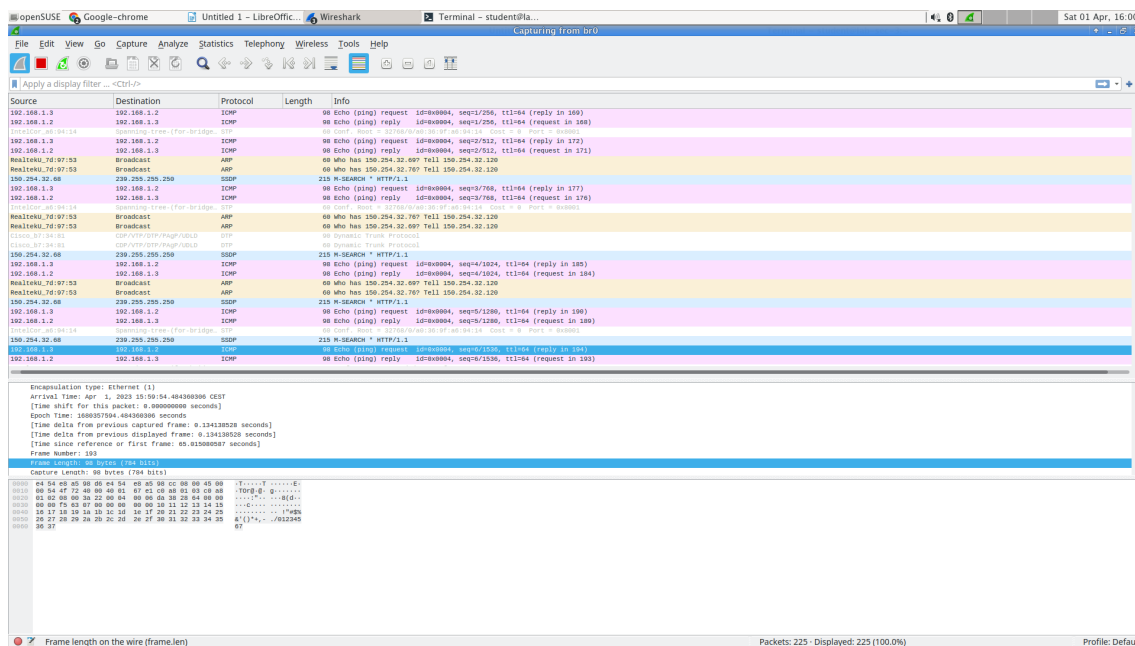
192.168.X.Y, gdzie X to nr rzędu, zaś Y to nr komputera

192.168.1.0 / 24

1.2 Przypisz jeden z adresów do interfejsu em1 (każdy komputer w rzędzie ma mieć inny adres).

```
lab-sec-3:/homex/student # ip addr add 192.168.1.3/24 dev br0
```

1.3 Przeanalizuj ruch na interfejsie em1 przy pomocy programu wireshark.



Rysunek 1: Ruch na interfejsie br0.

1.4 Zbadaj przy pomocy polecenia ping połączenie z innymi komputerami w rzędzie.

```
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=0.545 ms
```

```
64 bytes from 192.168.1.2: icmp_seq=2 ttl=64 time=0.525 ms
```

```
64 bytes from 192.168.1.2: icmp_seq=3 ttl=64 time=0.556 ms
```

```
^C
```

```
--- 192.168.1.2 ping statistics ---
```

Połączenie z innym komputerem w rzędzie działa.

1.5 Zbadaj przy pomocy polecenia ping połączenie komputerami w innych rzędach.

```
lab-sec-3:/homex/student # ping 192.168.2.5
```

```
PING 192.168.2.5 (192.168.2.5) 56(84) bytes of data.
From 150.254.4.66 icmp_seq=33 Destination Net Unreachable
From 150.254.4.66 icmp_seq=34 Destination Net Unreachable
```

Połączenie z komputerami w pozostałych rzędach jest przy obecnej konfiguracji jest niewykonalne, ponieważ utworzyliśmy . Można wykonać połączenie za pomocą obecnie skonfigurowanego adresu IP:

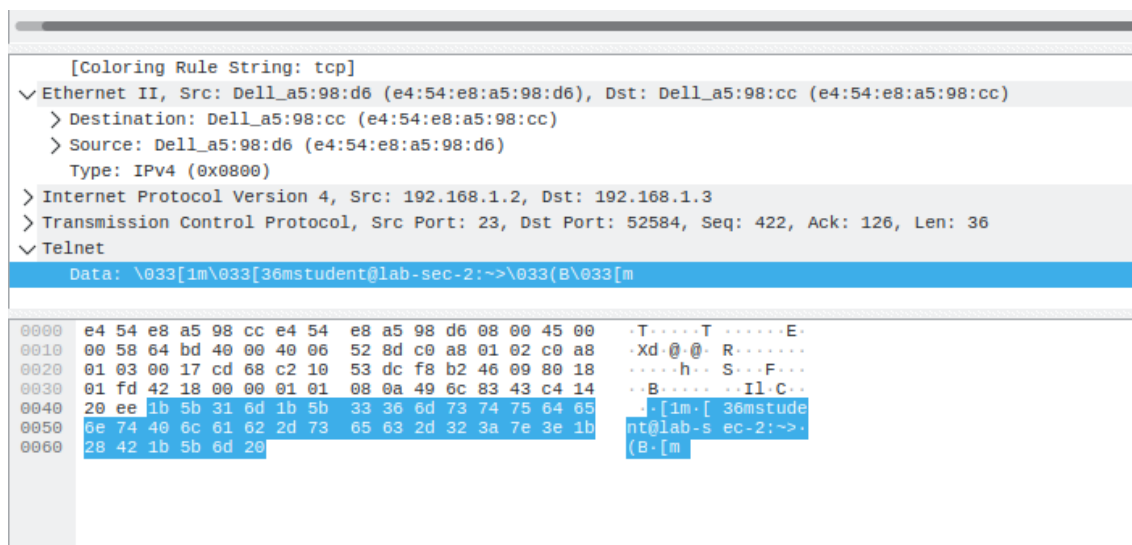
```
lab-sec-3:/homex/student # ping 150.254.32.70
PING 150.254.32.70 (150.254.32.70) 56(84) bytes of data.
64 bytes from 150.254.32.70: icmp_seq=1 ttl=64 time=0.938 ms
64 bytes from 150.254.32.70: icmp_seq=2 ttl=64 time=0.370 ms
64 bytes from 150.254.32.70: icmp_seq=3 ttl=64 time=0.613 ms
64 bytes from 150.254.32.70: icmp_seq=4 ttl=64 time=0.620 ms
64 bytes from 150.254.32.70: icmp_seq=5 ttl=64 time=0.618 ms
64 bytes from 150.254.32.70: icmp_seq=6 ttl=64 time=0.730 ms
```

1.6 Parami połącz się z innym komputerem z rzędu przy pomocy komendy telnet i zbadaj treść wymienianych pakietów.

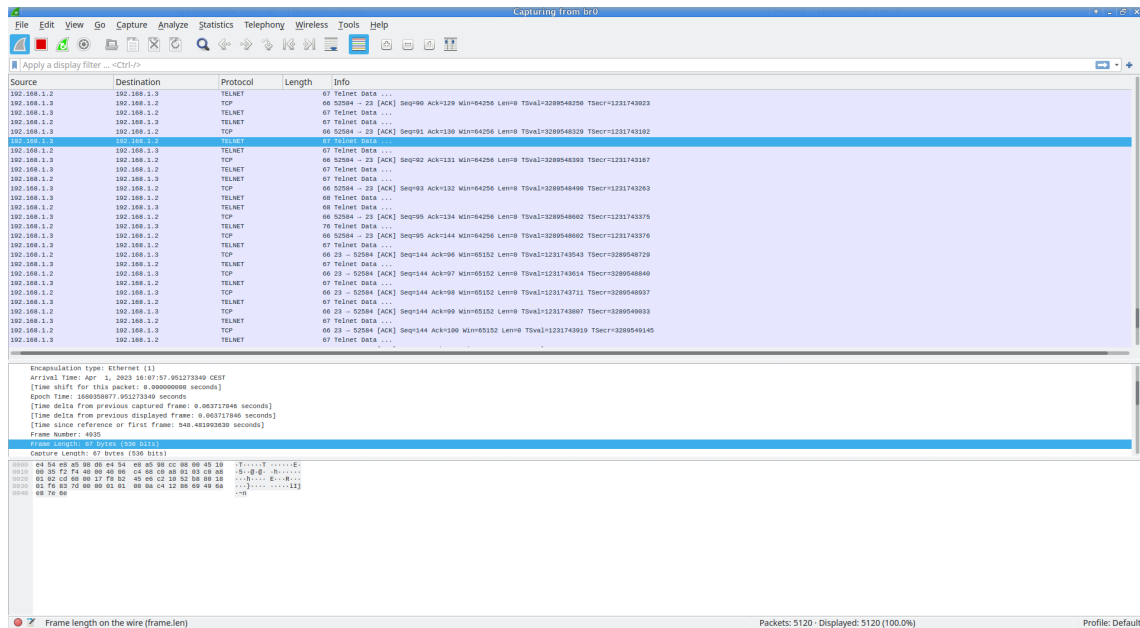
```
lab-sec-3:/homex/student # telnet 192.168.1.2
Trying 192.168.1.2...
Connected to 192.168.1.2.
Escape character is '^]'.
```

```
Linux 5.14.21-150400.24.46-default (lab-sec-2) (2)
```

```
lab-sec-2 login: student
Password:
Last login: Sat Apr 1 16:04:22 CEST 2023 from ::ffff:192.168.1.3 on pts/1
Last login: Sat Apr 1 16:07:59 from ::ffff:192.168.1.3
Have a lot of fun...
student@lab-sec-2:~> whoami
student
```



Rysunek 2: Zawartość pakietu telnet.



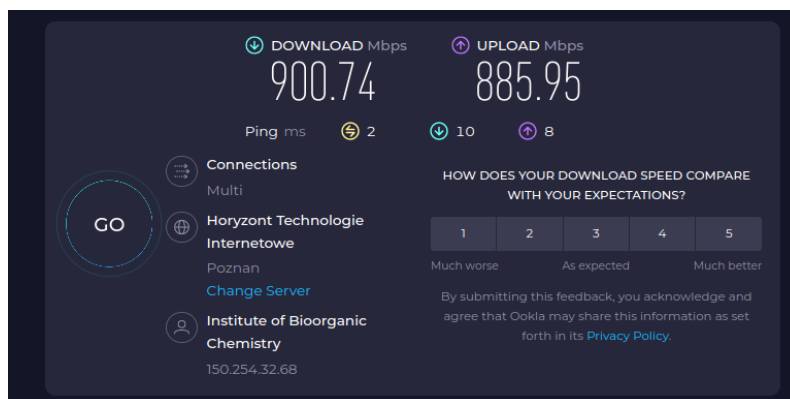
Rysunek 3: Wyniki pomiaru prędkości

Możemy zauważyć, że telnet przesyła nieszyfrowane dane, które z łatwością możemy odczytać za pomocą programu Wireshark.

2 Zadanie 2

W zadaniu pracuję na interfejsie em2.

2.1 Zmierz osiąganą maksymalną prędkość transmisji przy użyciu usługi speedtest.net.



Rysunek 4: Wyniki pomiaru prędkości

Aktualnie interfejs ustawiony jest w tryb pełnego duplexu przy prędkości 1000 Mb/s.

```
lab-sec-3:/homex/student # ethtool em2
Settings for em2:
Supported ports: [ TP ]
Supported link modes:   10baseT/Half 10baseT/Full
                        100baseT/Half 100baseT/Full
                        1000baseT/Full
Supported pause frame use: No
```

```
Supports auto-negotiation: Yes
Supported FEC modes: Not reported
Advertised link modes: 10baseT/Half 10baseT/Full
                      100baseT/Half 100baseT/Full
                      1000baseT/Full
Advertised pause frame use: No
Advertised auto-negotiation: Yes
Advertised FEC modes: Not reported
Speed: 1000Mb/s
Duplex: Full
Auto-negotiation: on
Port: Twisted Pair
PHYAD: 2
Transceiver: internal
MDI-X: off (auto)
Supports Wake-on: pumbg
Wake-on: g
      Current message level: 0x00000007 (7)
                          drv probe link

Link detected: yes
```

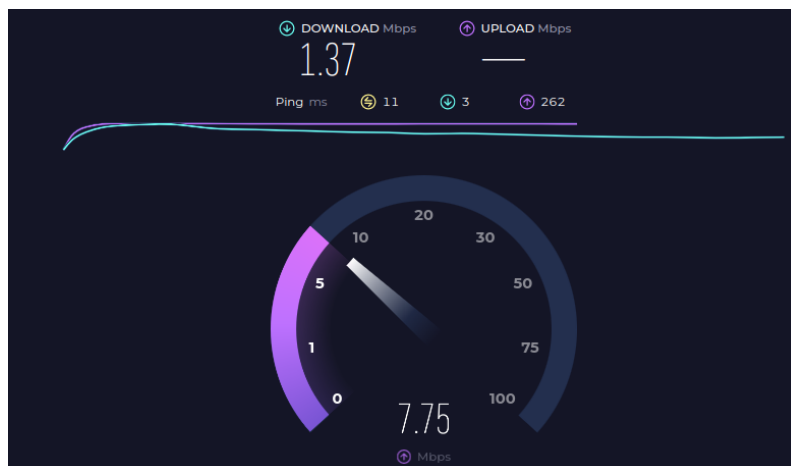
2.2 Zmień parametry em1 ograniczając prędkość transmisji i zmieniając parametry duplexu.

Modyfikuję prędkość interfejsu do 10 Mb/s w trybie pół-dupleksowym.

```
lab-sec-3:/homex/student # ethtool -s em2 speed 10 duplex half
```

2.3 Ponownie zmierz maksymalną prędkość transmisji. Wróć do pierwotnych ustawień. Przy pomocy odpowiedniego parametru komendy ethtool zidentyfikuj interfejsy p4p1 oraz p4p2.

Możemy zaobserwować znaczący spadek:



Rysunek 5: Wyniki pomiaru prędkości

Przywracam ustawienia interfejsu:

```
lab-sec-3:/homex/student # ethtool -s em2 speed 1000 duplex full
lab-sec-3:/homex/student # ethtool -p p4p1
```

3 Zadanie 3

3.1 Usuń nadane wcześniej adresy IP na em1 (mają pozostać tylko standardowe adresy, tj. uzyskiwane przez DHCP).

Usuwać wcześniej nadany adres IP:

```
lab-sec-3:/homex/student # ip address del 192.168.1.3/24 dev br0
```

3.2 Podłącz swój komputer poprzez port p4p1 do przełącznika na zapleczu (wszystkie komputery są podłączane do tego samego przełącznika).

Podłączyłem swój komputer pod portem nr 99 pod switch, poprzez port p4p1.

3.3 Skonfiguruj adres IP na porcie p4p1, tak by wszystkie komputery w laboratorium mogły komunikować się między sobą poprzez nowo utworzoną sieć

Wykonałem następującą komendę:

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

Komunikacja działa zarówno w obrębie rzędu jak i sali:

```
lab-sec-3:/homex/student # ip addr add 192.168.1.3/16 dev p4p1
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=0.549 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=64 time=0.714 ms
64 bytes from 192.168.1.2: icmp_seq=3 ttl=64 time=0.250 ms
^C
--- 192.168.1.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2040ms
rtt min/avg/max/mdev = 0.250/0.504/0.714/0.192 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=0.757 ms
64 bytes from 192.168.2.5: icmp_seq=2 ttl=64 time=0.706 ms
64 bytes from 192.168.2.5: icmp_seq=3 ttl=64 time=0.693 ms
^C
--- 192.168.2.5 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2056ms
rtt min/avg/max/mdev = 0.693/0.718/0.757/0.027 ms
```

Możemy zaobserwować komunikację w obrębie sieci, m.in. protokół ARP:

Source	Destination	Protocol	Length	Info
Broadcom_ba:13:04	Broadcast	ARP	60	Who has 192.168.5.13? Tell 192.168.3.10
Broadcom_ba:13:04	Broadcast	ARP	60	Who has 192.168.5.13? Tell 192.168.3.10
Broadcom_ba:13:04	Broadcast	ARP	60	Who has 192.168.5.13? Tell 192.168.3.10
Broadcom_ba:13:04	Broadcast	ARP	60	Who has 192.168.5.13? Tell 192.168.3.10
Cisco_35:40:01	Cisco_35:40:01	LOOP	0	Reply
Broadcom_ba:13:04	Broadcast	ARP	60	Who has 192.168.5.13? Tell 192.168.3.10
Broadcom_ba:13:04	Broadcast	ARP	60	Who has 192.168.5.13? Tell 192.168.3.10
192.168.1.3	192.168.2.5	ICMP	60	Echo (ping) request seq=7256, ttl=64 (reply in 10)
192.168.2.5	192.168.1.3	ICMP	60	Echo (ping) reply seq=7256, ttl=64 (request in 9)
Broadcom_ba:13:04	Broadcast	ARP	60	Who has 192.168.5.13? Tell 192.168.3.10
192.168.1.3	192.168.2.5	ICMP	60	Echo (ping) request seq=7512, ttl=64 (reply in 13)
192.168.2.5	192.168.1.3	ICMP	60	Echo (ping) reply seq=7512, ttl=64 (request in 12)
Broadcom_ba:13:04	Broadcast	ARP	60	Who has 192.168.5.13? Tell 192.168.3.10
192.168.1.3	192.168.2.5	ICMP	60	Echo (ping) request seq=7768, ttl=64 (reply in 16)
Broadcom_ba:13:04	Broadcast	ARP	60	Who has 192.168.5.13? Tell 192.168.3.10
192.168.1.3	192.168.2.5	ICMP	60	Echo (ping) request seq=8024, ttl=64 (reply in 19)
192.168.2.5	192.168.1.3	ICMP	60	Echo (ping) reply seq=8024, ttl=64 (request in 18)
Broadcom_ba:13:04	Broadcast	ARP	60	Who has 192.168.5.13? Tell 192.168.3.10
192.168.1.3	192.168.2.5	ICMP	60	Echo (ping) request seq=8280, ttl=64 (reply in 22)
192.168.2.5	192.168.1.3	ICMP	60	Echo (ping) reply seq=8280, ttl=64 (request in 21)
Broadcom_ba:13:04	Broadcast	ARP	60	Who has 192.168.5.13? Tell 192.168.3.10
Broadcom_ba:13:7c	Broadcom_ba:13:7c	ARP	42	Who has 192.168.2.5? Tell 192.168.1.3
Broadcom_ba:13:08	Broadcom_ba:13:7c	ARP	60	192.168.2.5 is at 00:13:13:ba:13:08
192.168.1.3	192.168.2.5	ICMP	60	Echo (ping) request seq=8536, ttl=64 (reply in 27)
192.168.2.5	192.168.1.3	ICMP	60	Echo (ping) reply seq=8536, ttl=64 (request in 26)

Frame 1: 60 Bytes on wire (480 bits), 60 Bytes captured (480 bits) on interface p4p1, id 0
> Interface id: 0 (p4p1)
Encapsulation type: Ethernet (1)
Arrival Time: Apr 1, 2023 16:36:53.415518888 CEST
[Time shift for this packet: 0.000000000 seconds]
Epoch Time: 1680388813.415518888 seconds
[Time delta from previous captured frame: 0.000000000 seconds]
[Time delta from previous displayed frame: 0.000000000 seconds]
[Time since reference or first frame: 0.000000000 seconds]
Frame Number: 1
0000 ff ff ff ff ff ff 3b 8c 13 04 08 06 00 01
0010 00 00 00 04 00 01 23 15 00 13 04 c0 a8 03 0a
0020 00 00 00 00 00 00 c0 a8 03 04 00 00 00 00
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00

p4p1: <live capture in progress> Packets: 40 · Displayed: 40 (100.0%) Profile: Default

Rysunek 6: Wyniki pomiaru prędkości