

1. Skonfiguruj 4 maszyny wirtualne Virbian1 – Virbian4, tak aby korzystały z sieci local0 i local1, tak jak zaznaczono na rysunku powyżej. Nazwij ich interfejsy tak, jak na rysunku (enp0, enp1 i enp-out).

W zakładce sieci wybieramy następujące opcje dla poszczególnych maszyn:

- *Virbian1:*
 - *Adapter 1: Attached to: Internal Network Name: local0*
- *Virbian2:*
 - *Adapter 1: Attached to: Internal Network Name: local0*
- *Virbian3:*
 - *Adapter 1: Attached to: Internal Network Name: local0*
 - *Adapter 2: Attached to: Internal Network Name: local1*
- *Virbian4:*
 - *Adapter 2: Attached to: Internal Network Name: local1*

Nazywamy interfejsy poszczególnych maszyn (informację o tym, jaki interfejs na danej maszynie odpowiada jakiej sieci możemy uzyskać porównując adresy MAC z interfejsu i z sekcji ustawień sieci maszyny):

- *Virbian1:*
 - *sudo ip link set enp0s3 name enp0*
- *Virbian2:*
 - *sudo ip link set enp0s3 name enp0*
- *Virbian3:*
 - *sudo ip link set enp0s3 name enp0*
 - *sudo ip link set enp0s8 name enp1*
- *Virbian4:*
 - *sudo ip link set enp0s8 name enp-out*

2. Przypisz trzem interfejsom podłączonym do wirtualnej sieci local0 adresy z sieci 192.168.1.0/24 takie jak na rysunku. Przypisz dwóm interfejsom podłączonym do wirtualnej sieci local1 adresy z sieci 192.168.2.0/24 takie jak na rysunku.

Na każdej maszynie dany interfejs najpierw aktywujemy, a następnie przypisujemy adres wybrany zgodnie z poleceniem:

- *Virbian1:*
 - *sudo ip link set up dev enp0*
 - *sudo ip addr add 192.168.1.1/24 dev enp0*
- *Virbian2:*
 - *sudo ip link set up dev enp0*
 - *sudo ip addr add 192.168.1.2/24 dev enp0*
- *Virbian3:*
 - *sudo ip link set up dev enp0*

- *sudo ip addr add 192.168.1.3/24 dev enp0*
- *sudo ip link set up dev enp1*
- *sudo ip addr add 192.168.2.1/24 dev enp1*
- Virbian4:
 - *sudo ip link set up dev enp-out*
 - *sudo ip addr add 192.168.2.2/24 dev enp-out*

3. Poleceniem ping sprawdź wzajemną osiągalność maszyn podłączonych do tej samej sieci local0 i maszyn podłączonych do tej samej sieci local1.

Virbian1:

```

Virbian1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
user@virbian:~$ 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=1.49 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=64 time=0.851 ms
64 bytes from 192.168.1.2: icmp_seq=3 ttl=64 time=1.06 ms
^C
--- 192.168.1.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 6ms
rtt min/avg/max/mdev = 0.851/1.133/1.489/0.266 ms
user@virbian:~$ ping 192.168.1.3
PING 192.168.1.3 (192.168.1.3) 56(84) bytes of data.
64 bytes from 192.168.1.3: icmp_seq=1 ttl=64 time=1.47 ms
64 bytes from 192.168.1.3: icmp_seq=2 ttl=64 time=1.17 ms
64 bytes from 192.168.1.3: icmp_seq=3 ttl=64 time=0.989 ms
^C
--- 192.168.1.3 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 6ms
rtt min/avg/max/mdev = 0.989/1.209/1.467/0.196 ms
user@virbian:~$ _

```

Virbian2:

```

Virbian2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
user@virbian:~$ ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=0.775 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=1.00 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=1.00 ms
^C
--- 192.168.1.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 5ms
rtt min/avg/max/mdev = 0.775/0.925/1.001/0.112 ms
user@virbian:~$ ping 192.168.1.3
PING 192.168.1.3 (192.168.1.3) 56(84) bytes of data.
64 bytes from 192.168.1.3: icmp_seq=1 ttl=64 time=1.88 ms
64 bytes from 192.168.1.3: icmp_seq=2 ttl=64 time=1.17 ms
64 bytes from 192.168.1.3: icmp_seq=3 ttl=64 time=1.10 ms
^C
--- 192.168.1.3 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 5ms
rtt min/avg/max/mdev = 1.095/1.380/1.877/0.353 ms
user@virbian:~$

```

Virbian3:

```
Virbian3 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
user@virbian:~$ ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=0.650 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=0.798 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=1.17 ms
^C
--- 192.168.1.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 4ms
rtt min/avg/max/mdev = 0.650/0.873/1.173/0.222 ms
user@virbian:~$ 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=0.776 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=64 time=1.14 ms
64 bytes from 192.168.1.2: icmp_seq=3 ttl=64 time=1.41 ms
^C
--- 192.168.1.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 5ms
rtt min/avg/max/mdev = 0.776/1.109/1.414/0.261 ms
user@virbian:~$ ping 192.168.2.2
PING 192.168.2.2 (192.168.2.2) 56(84) bytes of data.
64 bytes from 192.168.2.2: icmp_seq=1 ttl=64 time=1.44 ms
64 bytes from 192.168.2.2: icmp_seq=2 ttl=64 time=0.758 ms
64 bytes from 192.168.2.2: icmp_seq=3 ttl=64 time=1.01 ms
^C
--- 192.168.2.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 5ms
rtt min/avg/max/mdev = 0.758/1.069/1.444/0.283 ms
user@virbian:~$
```

Virbian4:

```
Virbian4 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
user@virbian:~$ ping 192.168.2.1
PING 192.168.2.1 (192.168.2.1) 56(84) bytes of data.
64 bytes from 192.168.2.1: icmp_seq=1 ttl=64 time=0.831 ms
64 bytes from 192.168.2.1: icmp_seq=2 ttl=64 time=0.871 ms
64 bytes from 192.168.2.1: icmp_seq=3 ttl=64 time=1.03 ms
^C
--- 192.168.2.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 5ms
rtt min/avg/max/mdev = 0.831/0.909/1.026/0.087 ms
user@virbian:~$ _
```

4. Na maszynach Virbian1, Virbian2 i Virbian4 dodaj trasy domyślne, które na rysunku powyżej zaznaczone są przerywanymi strzałkami. Przykładowo trasa domyślna z maszyny Virbian2 powinna prowadzić przez adres 192.168.1.3.

Dla podanych maszyn dodajemy wpisy w tablicy routingu określające trasę domyślną za pomocą komend:

- Virbian1:
 - `sudo ip route add default via 192.168.1.2`
- Virbian2:
 - `sudo ip route add default via 192.168.1.3`
- Virbian4:
 - `sudo ip route add default via 192.168.2.1`

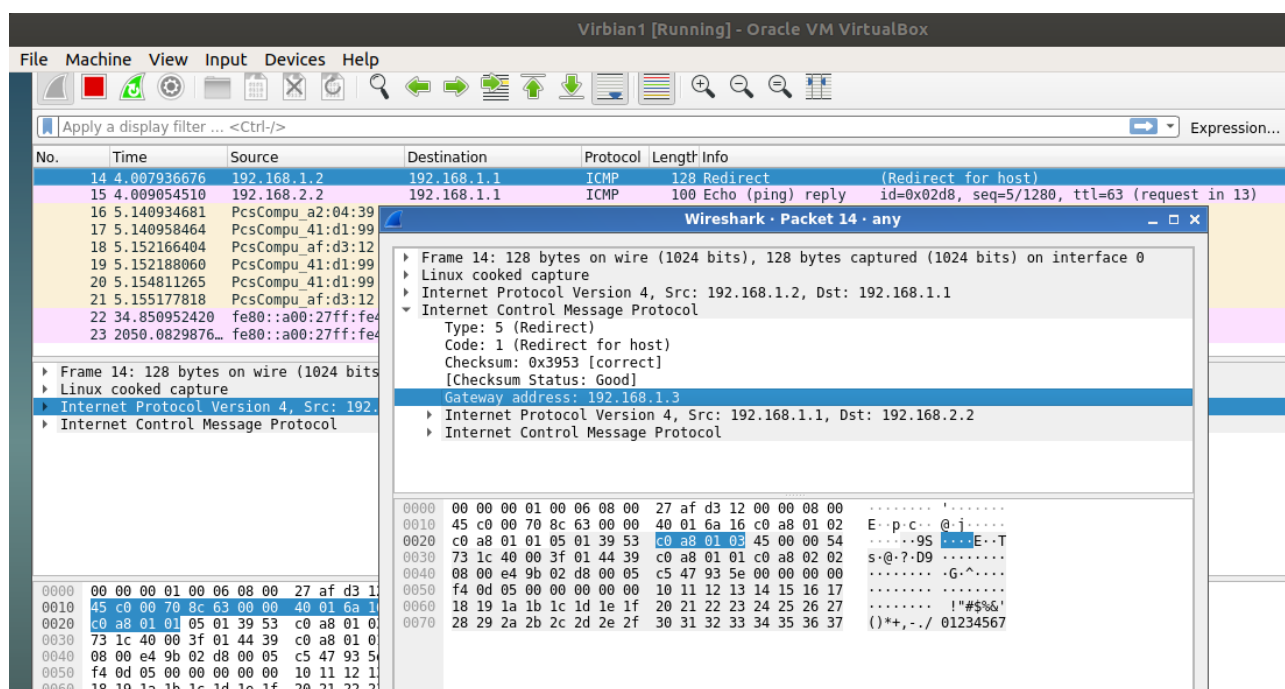
5. Włącz Wireshark na wszystkich maszynach. Następnie z maszyny Virbian1 pingnij maszynę Virbian4. Zaobserwuj, że maszyna jest osiągalna, ale oprócz komunikatów ICMP reply maszyna Virbian1 otrzymuje również komunikaty ICMP redirect. Są one wysyłane przez maszynę Virbian2 i informują o tym, że routing na maszynie Virbian1 jest prawdopodobnie źle skonfigurowany. Odpowiedz na następujące pytania:

- Jaka jest sugerowana przez maszynę Virbian2 modyfikacja tablicy routingu na maszynie Virbian1 ?
- Dlaczego taka zmiana ma sens?
- W jaki sposób maszyna Virbian2 mogła wykryć powyższy problem?

Wykonujemy ping maszyny Virbian4 z maszyny Virbian1. Zauważamy, że otrzymujemy nie tylko odpowiedź, ale również komunikaty ICMP redirect od 192.168.1.2, czyli Virbian2:

The screenshot displays a VirtualBox VM environment. The top window shows the Wireshark network protocol analyzer with a packet capture on the 'eth0' interface. The packet list shows several ICMP Echo (ping) requests and replies, as well as ICMP Redirect messages. The packet details pane shows the 'Internet Protocol Version 4 (ip), 20 bytes' section. The bottom window shows a terminal session where a ping command is being executed: `user@virbian:~$ ping 192.168.2.2`. The output shows several ICMP Redirect messages from 192.168.1.2 to 192.168.2.2, indicating a routing issue. The output also shows the ping statistics: `5 packets transmitted, 5 received, 0% packet loss, time 11ms`.

Maszyna *Virbian2* sugeruje kierowanie ruchu z maszyny *Virbian1* do sieci *local1* bezpośrednio przez maszynę *Virbian3*:



W naszym wypadku oznaczałoby to zmianę trasy domyślnej na maszynie *Virbian1* z 192.168.1.2 na 192.168.1.3, co moglibyśmy zrealizować przy pomocy następujących poleceń w terminalu *Virbian1*:

- `sudo ip route del default`
- `sudo ip route add default via 192.168.1.3`

Zmiana ta jest sensowna, gdyż skróciłaby trasę pakietu o jeden.

Zamiast:

- *Virbian1* -> *Virbian2* -> *Virbian3* -> *Virbian4*

Mielibyśmy:

- *Virbian1* -> *Virbian3* -> *Virbian4*

W jaki sposób maszyna *Virbian2* mogła wykryć zgłoszony przez nią problem? W momencie otrzymania pakietu ICMP od *Virbian1*, zostaje dopasowany adres bramy z tablicy routingu, pod który pakiet zostanie przesłany dalej. Następnie, maszyna *Virbian2* sprawdza, czy adres nadawcy i dopasowany adres bramy należą do tej samej sieci. Jeśli tak jest, wówczas wysyła informację zwrotną w postaci pakietu *ICMP Redirect* do nadawcy wraz z dopasowanym adresem bramy, aby nadawca mógł zaktualizować swoją tablicę routingu o ten adres.