

# Warsztaty z Sieci komputerowych

## Lista 2

### Przed zajęciami

Utwórz cztery maszyny *Virbian1–Virbian4*, każdą z dwiema kartami sieciowymi. Niech pierwsze karty sieciowe maszyn *Virbian1* i *Virbian2* będą połączone wirtualną siecią **remote**, zaś ich drugie karty podłączone do wirtualnych sieci **local1** i **local2** (*Virbian1* do **local1** a *Virbian2* do **local2**). Maszyn *Virbian3* i *Virbian4* na razie nie konfiguruj. Uruchom maszyny *Virbian1* i *Virbian2*.

### Tutorial #1

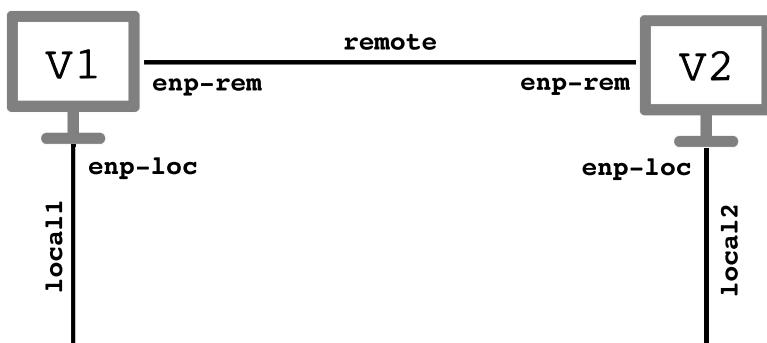
Po uruchomieniu maszyn *Virbian1* i *Virbian2* na każdej z nich dostępne powinny być trzy interfejsy sieciowe (możesz wyświetlić je poleceniem `ip link`): interfejs pętli lokalnej `lo` i dwa interfejsy o nazwach zaczynających się od `enp`.

- Na maszynie *Virbian1* ustal interfejs `enpxsy` połączony z siecią **remote** oraz interfejs `enpwsz` połączony połączony z siecią **local1**. Dla wygody zmienimy nazwy tych interfejsów poleceniami:

```
V1#> ip link set enpxsy name enp-rem  
V1#> ip link set enpwsz name enp-loc
```

1. `lo`  
2. `remote`  
3. `localN`

Wykonaj analogiczne polecenia na maszynie *Virbian2* zmieniając nazwy jej interfejsów połączonych z sieciami **remote** i **local2** odpowiednio na `enp-rem` i `enp-loc`. Otrzymasz sytuację jak na poniższym rysunku.



- Poleceniem `ip link` aktywuj karty `enp-rem` a poleceniem `ip addr` przypisz im adresy IP. Na maszynie *Virbian1* przypisz tej karcie adres IP równy `172.16.0.1/16`, zaś na maszynie *Virbian2* adres `172.16.0.2/16`. Poleceniem `ip addr` wyświetl aktualnie skonfigurowane interfejsy. W przypadku błędu wykorzystaj polecenie `ip addr flush`.

# `ip link set up dev enp-rem`  
# `ip addr add [ip]/16 dev enp-rem`

- Wyświetl tablicę routingu poleceniem

V1\$> ip route

a także wykorzystując starsze narzędzie route:

V1\$> route -n

Powyższe adresy IP	Klasa	Ilość klas
10.0.0.0/8	A	1
172.16.0.0/12	B	16
192.168.0.0/16	C	256

Zauważ, że polecenie `ip route` wyświetla przy trasach do sieci opis proto kernel. Oznacza to, że trasa do danej sieci została dodana automatycznie przez jądro systemu podczas dodawania adresu do interfejsu.

- Za pomocą programu ping sprawdź, czy połączone ze sobą siecią remote maszyny wirtualne *Virbian1* i *Virbian2* „widzą się” wzajemnie.
- Pinguj adres interfejsu pętli lokalnej 10 o adresie 127.0.0.1. Zauważ, że komunikaty dochodzą pomimo tego, że odpowiedni wpis nie jest wyświetlany poleceniem `ip route`. Te dodatkowe wpisy w tablicy można wyświetlić poleceniem

V1\$> ip route list table local

### broadcast

Przeanalizuj poszczególne wiersze. Zwróć uwagę na adresy rozgłoszeniowe i różnice w polach scope.

- Na maszynach *Virbian1* i *Virbian2* włącz Wiresharka i rozpoczęj nasłuchiwanie na wszystkich interfejsach. Zaobserwuj, co jest wypisywane w konsoli oraz jakie pakiety są wysyłane i odbierane jeśli pingasz z maszyny *Virbian1*:
  - adres 127.0.0.1; **V1: widać pakiety V2: nic nie ma**
  - swój własny adres IP przypisany do interfejsu `enp-rem`; **V1: widać V2: nic**
  - adres IP maszyny *Virbian2* przypisany do interfejsu `enp-rem`; **V1: widać V2: widać**
  - adres rozgłoszeniowy sieci podłączonej do interfejsu `enp-rem` (poleceniem `ping -b 172.16.255.255`)<sup>1</sup>; **WARNING: Ping to broadcast address (\*)**
  - adres IP należący do sieci 172.16.0.0/16 nieprzypisany do żadnego interfejsu sieciowego; **np. 172.16.100.200 → Who has [ip]? Tell 172.16.0.1, ARP,**
  - adres IP z sieci, do której maszyna nie jest podłączona, np. 10.10.10.10. **(obie VM)**

Porównaj otrzymane komunikaty, przesyłane pakiety i czasy reakcji.

↪ W Wiresharku nic, w konsoli "connect: Network is unreachable" ↩ w V1 (konsola) Destination unreachable

## Tutorial #2

Skonfiguruj maszyny *Virbian3* i *Virbian4*, które będą miały po jednej karcie sieciowej połączonej odpowiednio z siecią `local1` i `local2`. Po ich uruchomieniu zmień nazwę wirtualnej karty sieciowej w tych maszynach na `enp-loc`. Dzięki temu osiągniesz konfigurację jak na poniższym rysunku.

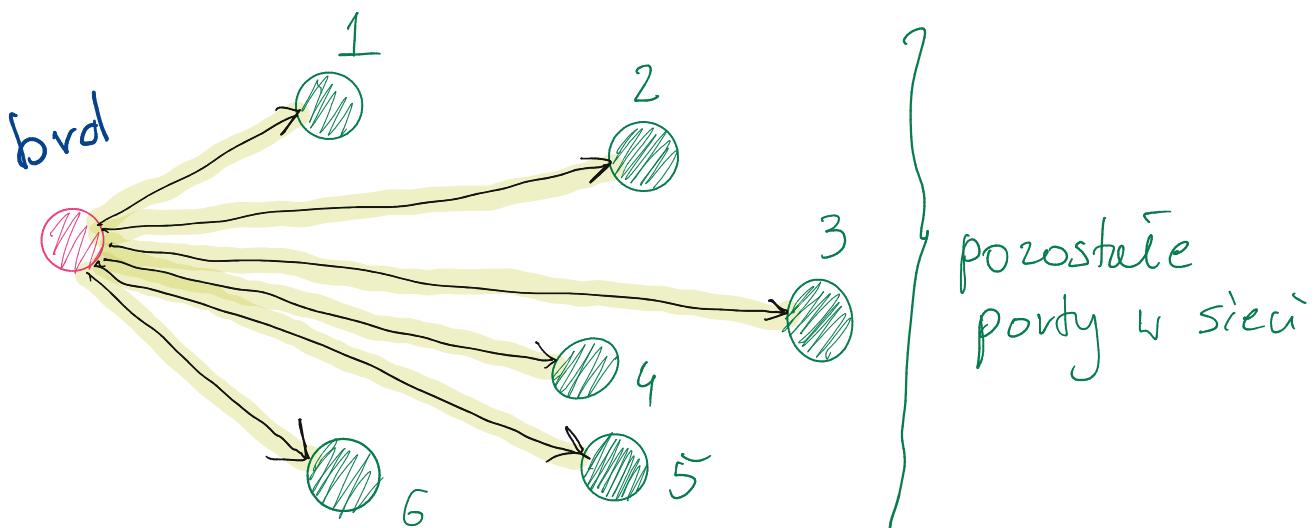
**V3 → local1      V4 → local2**

<sup>1</sup> Jeśli nie otrzymujesz odpowiedzi, na maszynie docelowej sprawdź wynik działania polecenia `sysctl net.ipv4.icmp_echo_ignore_broadcasts`. Jeśli jest on równy 1, to odbierane pingi wysłane na adres rozgłoszeniowy będą ignorowane. W takim wypadku należy wykonać polecenie `sysctl net.ipv4.icmp_echo_ignore_broadcasts=0`.

# ip link set enp0s3 name enp-loc

Gdyby się okazało, że w normalnej sieci odpowiadająca ping byłaby zduplikowana, to nasze IP byłoby używane przez więcej niż jedną maszynę = nasza sieć byłaby źle skonfigurowana.

broadcast → wysyłanie przez jeden port pakietów, które powinny być odebrane przez wszystkie pozostałe porty przyłączone do sieci



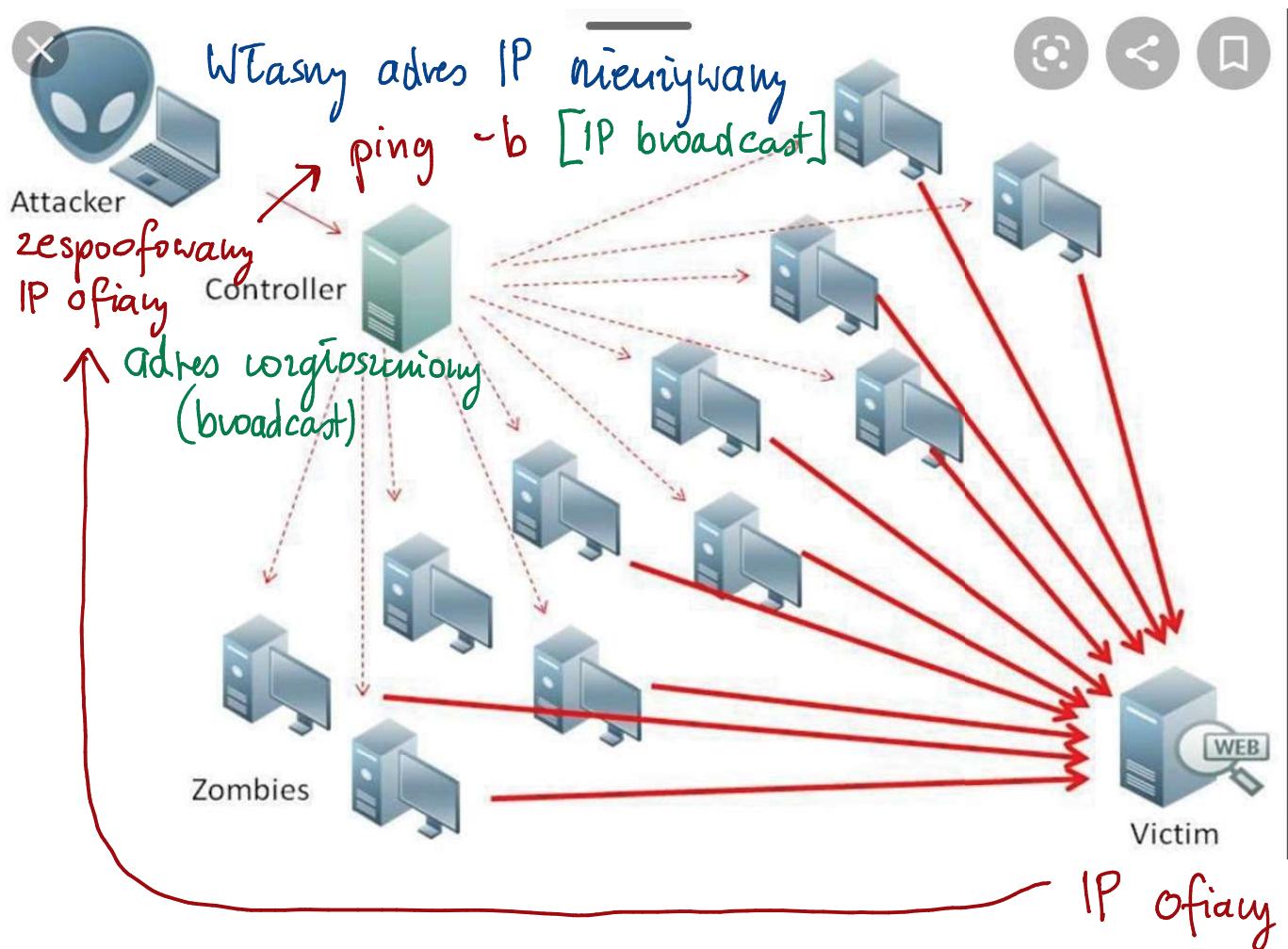
```
user@virbian:~$ ip route list table local
broadcast 127.0.0.0 dev lo proto kernel scope link src 127.0.0.1
local 127.0.0.0/8 dev lo proto kernel scope host src 127.0.0.1
local 127.0.0.1 dev lo proto kernel scope host src 127.0.0.1
broadcast 127.255.255.255 dev lo proto kernel scope link src 127.0.0.1
broadcast 172.16.0.0 dev enp-rem proto kernel scope link src 172.16.0.1
local 172.16.0.1 dev enp-rem proto kernel scope host src 172.16.0.1
broadcast 172.16.255.255 dev enp-rem proto kernel scope link src 172.16.0.1
```

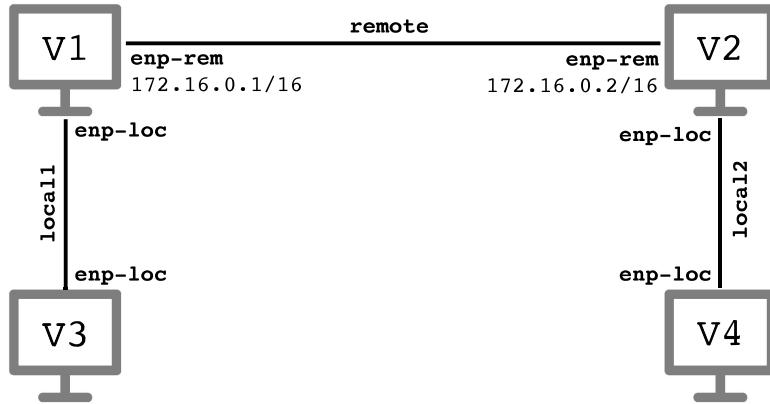
inne sieci nie są skonfigurowane (np. enp-loc)  
 local → scope host → adres poprawny tylko w obiektie hosta  
 broadcast → scope link → adres poprawny tylko w obiektie warstwy fizycznej (ethernet, WiFi)

Przykładowe brd:

- V1: request z 172.16.0.1 do brd (172.16.255.255)  
 2x reply z 172.16.0.1 i 172.16.0.2 do 172.16.0.1  
 w konsoli pokazuje „DUP”
- V2: request z 172.16.0.1 do 172.16.0.2, reply vice versa

Smurf attack → atak typu ping flood polegający na wysyłaniu przez atakującego pingów ICMP na adres wyróżniony sieci. Atakujący musi oczywiście mieć sfalsowany adres IP (ten, który należy do ofiary).





Celem poniższego zadania jest konfiguracja routingu pomiędzy tymi maszynami wirtualnymi.

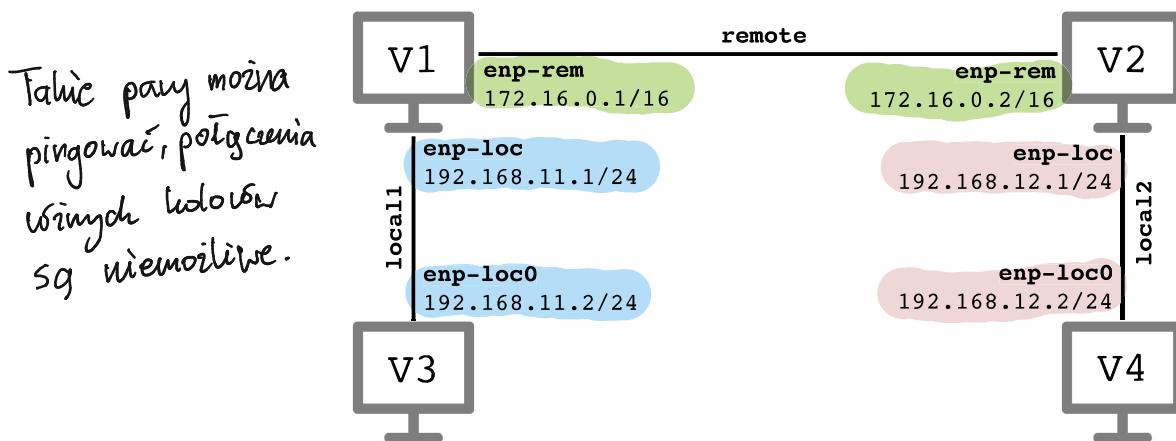
- ▶ Na wszystkich maszynach sprawdź wynik polecenia `sysctl net.ipv4.ip_forward` i jeśli zwracaną wartością jest 0, wykonaj polecenie  
WSZĘDZIEĆ BYŁO WŚWIĘTE 1  
`V#> sysctl net.ipv4.ip_forward=1`

Polecenie to włącza przekazywanie pakietów IP pomiędzy interfejsami (maszyna wirtualna będzie mogła pełnić funkcję routera).

- ▶ Aktywuj interfejsy `enp-loc` na maszynach *Virbian1* i *Virbian3* i przypisz im adresy IP równe odpowiednio `192.168.11.1/24` i `192.168.11.2/24`.

Aktywuj interfejsy `enp-loc` na maszynach *Virbian2* i *Virbian4* i przypisz im adresy IP równe odpowiednio `192.168.12.1/24` i `192.168.12.2/24`.

Uzyskana konfiguracja wygląda teraz jak na poniższym rysunku.



- ▶ Na każdej maszynie poleciem `ping` sprawdź osiągalność najbliższych interfejsów najbliżej położonych maszyn. Przykładowo na maszynie *Virbian1* należy sprawdzić osiągalność interfejsu `enp-loc0` maszyny *Virbian3* oraz interfejsu `enp-rem` maszyny *Virbian2* wydając polecenia

```
V1$> ping 192.168.11.2
V1$> ping 172.16.0.2
```

# ip link set up dev enp-loc  
# ip addr add [ip]/24 dev enp-loc

- Z maszyny *Virbian1* sprawdź osiągalność interfejsu `enp-loc` maszyny *Virbian2*, tzn. wydaj na niej polecenie `ping 192.168.12.1`. Analogicznie na maszynie *Virbian2* wydaj polecenie `ping 192.168.11.1`. Takie adresy są nieosiągalne, gdyż nadawca nie wie, jak dostać się do docelowej sieci (odpowiednio `192.168.12.0/24` i `192.168.11.0/24`).
- ✖ ► Spróbujmy to naprawić dodając na maszynach *Virbian1* i *Virbian2* odpowiednie wpisy w ich tablicach routingu. W tym celu wydaj polecenia:

```
V1#> ip route add 192.168.12.0/24 via 172.16.0.2
V2#> ip route add 192.168.11.0/24 via 172.16.0.1
```

Jeśli pomyliłeś się wpisując polecenie `ip route`, dodaną omyłkowo trasę możesz skasować zamieniając parametr `add` na `del`. Poleceniem `ip route` sprawdź, czy trasy zostały skonfigurowane. Wykonaj polecenia `ping` z poprzedniego punktu; teraz powinny zakończyć się sukcesem. *sos dwie nowe linijki + ping działa w obie strony*

- ✖ ► Z maszyny *Virbian1* sprawdź osiągalność interfejsu `enp-loc` maszyny *Virbian4* wydając na niej polecenie `ping 192.168.12.2`

Co jest przyczyną niepowodzenia? Jaki komunikat otrzymujesz? Obejrzyj przesyłane komunikaty ICMP Wiresharkami uruchomionymi na wszystkich maszynach. Zauważ, że komunikat *ICMP echo request* dociera do komputera docelowego. Dlaczego więc nie jest odsyłana odpowiednia odpowiedź?

- Spróbujmy naprawić sytuację dodając na maszynie *Virbian4* informację pozwalającą trafić do pozostałych dwóch sieci. Zamiast dodawać dwa wpisy do tablicy routingu, określmy, że pakiety mogą dotrzeć w dowolne miejsce, jeśli dostarczymy je do interfejsu `enp-loc` podłączonej bezpośrednio maszyny *Virbian2*. W tym celu wydaj polecenia

```
V4#> ip route add default via 192.168.12.1
```

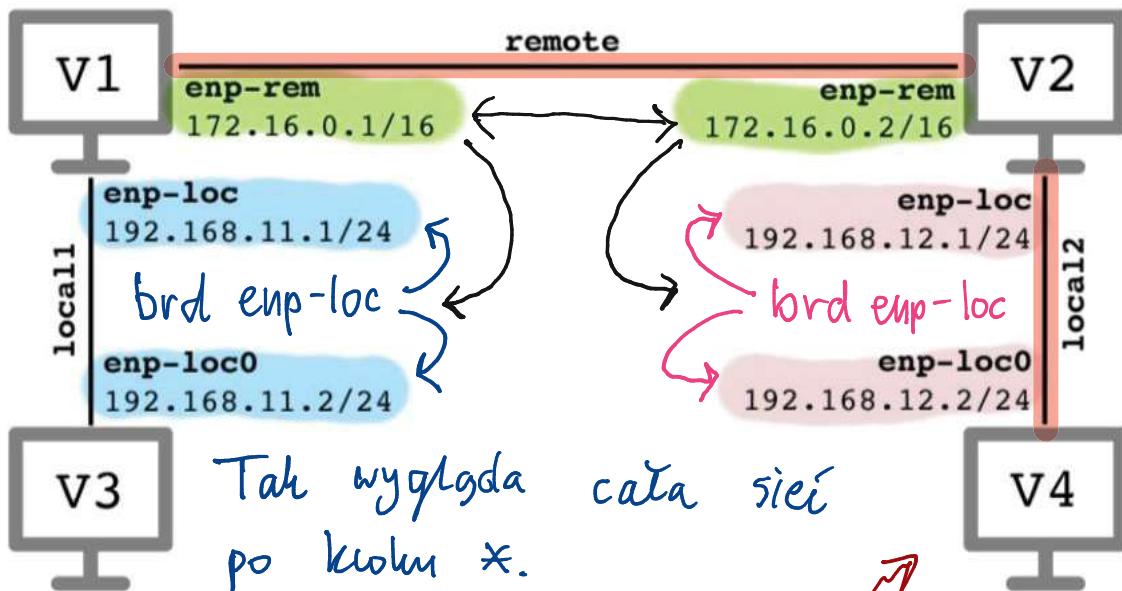
Napis `default` jest skrótem notacyjnym na `0.0.0.0/0`. Dodaj analogiczny wpis na maszynie *Virbian3*.

```
V3#> ip route add default via 192.168.11.1
```

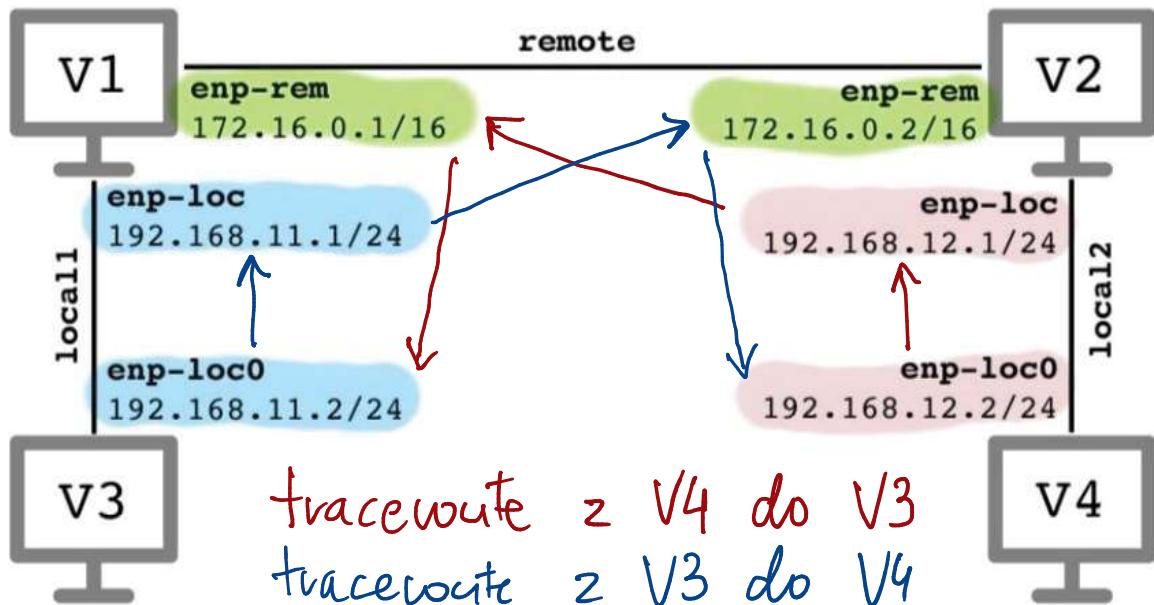
- Wyświetl bieżącą tablicę routingu na maszynach *Virbian3* i *Virbian4* poleceniem `ip route`. Sprawdź, że z każdej maszyny możesz teraz pingnąć adres dowolnego innego interfejsu. W razie kłopotów zaobserwuj przesyłane pakiety Wiresharkiem.
- Wyświetl trasę z maszyny *Virbian3* do maszyny *Virbian4* poleceniem `traceroute`. Analogicznie wyświetl trasę z maszyny *Virbian4* do maszyny *Virbian3*.
- Na wszystkich maszynach zdekonfiguruj istniejące interfejsy poleceniami `ip addr flush` i `ip link set down` a następnie wyłącz wszystkie cztery maszyny.

<i>Z</i>	<i>do</i>	<i>V<sub>1</sub></i>	<i>V<sub>1</sub></i> <i>rem</i>	<i>V<sub>2</sub></i>	<i>V<sub>2</sub></i> <i>rem</i>	<i>V<sub>2</sub></i> <i>loc</i>	<i>V<sub>3</sub></i>	<i>V<sub>4</sub></i>
<i>V<sub>1</sub></i>		✓	✓	✓	✓	✓	✓	✓
<i>V<sub>2</sub></i>		✓	✓	✓	✓	✓	✓	✓
<i>V<sub>3</sub></i>		✓	✓	✓	✓	✓	✓	✓
<i>V<sub>4</sub></i>		✓	✓	✓	✓	✓	✓	✓

<i>V<sub>1</sub> rem</i>	: 172.16.0.1 /16
<i>V<sub>1</sub> loc</i>	: 192.168.11.1 /24
<i>V<sub>2</sub> rem</i>	: 172.16.0.2 /16
<i>V<sub>2</sub> loc</i>	: 192.168.12.1 /24
<i>V<sub>3</sub></i>	: 192.168.11.2 /24
<i>V<sub>4</sub></i>	: 192.168.12.2 /24



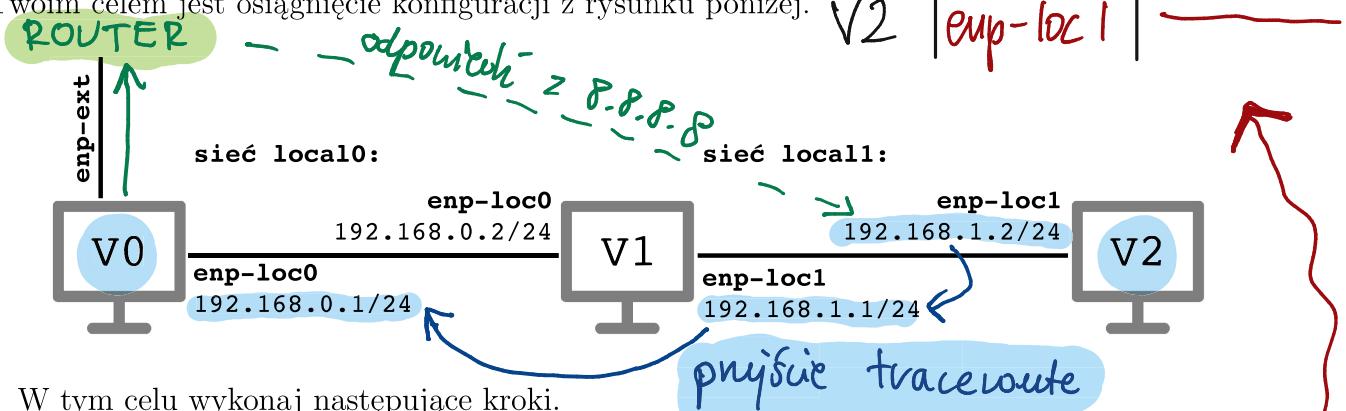
\*: V1 = ping 192.168.12.2  
 chyba o to dochodzi { Wysyłane są requesty, ale ARP nie potrafi zidentyfikować odbiorcy (adresu MAC). W pakietach ICMP widać w źródła (172.16.0.1) i docelowy (192.168.12.2). Może odpowiedzi nie dochodzi, bo jest trasa tylko w jednym kierunku, a więc po przejściu V1 → V2 → V4 nie może wrócić.



	enp0s3	enp0s8
V0	enp-ext	enp-loc 0
V1	enp-(loc)	enp-loc 1
V2	enp-loc 1	—

## Wyzwanie #1

Twoim celem jest osiągnięcie konfiguracji z rysunku poniżej.



W tym celu wykonaj następujące kroki.

- Utwórz maszyny wirtualne:

- *Virbian0*, która będzie miała dwie karty sieciowe: jedną z domyślną konfiguracją sieciową (NAT), zaś drugą połączoną z wirtualną siecią *local0*;
- *Virbian1*, która będzie miała dwie karty sieciowe połączoną z wirtualnymi sieciami *local0* i *local1*;
- *Virbian2*, która będzie miała jedną kartę sieciową połączoną z wirtualną siecią *local1*.

- Uruchom maszyny i nazwij ich interfejsy tak jak na rysunku powyżej. Uzyskaj konfigurację sieciową dla interfejsu *enp-ext* maszyny *Virbian0* polecением

v0#> dhclient -v enp-ext # ip link set [interfejs] name [nazwa]

- Przypisz obu interfejsom *enp-loc0* i obu interfejsom *enp-loc1* adresy IP z sieci 192.168.0.0/24 i 192.168.1.0/24 jak na rysunku. Sprawdź osiągalność interfejsów bezpośrednio połączonych maszyn polecieniem ping.

- Skonfiguruj routing:

- na maszynie *Virbian0* określ, że do sieci 192.168.1.0/24 można dostać się wysyłając pakiety do 192.168.0.2; # ip route add 192.168.1.0/24 via 192.168.0.2

- na maszynie *Virbian1* ustaw bramę domyślną równą 192.168.0.1.

- na maszynie *Virbian2* ustaw bramę domyślną równą 192.168.1.1;

- Z każdej maszyny wirtualnej powinny być teraz osiągalne wszystkie interfejsy *enp-loc*i**. Sprawdź to polecieniem ping. Wyświetl trasę z maszyny *Virbian2* do maszyny *Virbian0* polecieniem traceroute. V2\$: traceroute [-I] 192.168.0.1 nowe połączenie

- Z maszyny *Virbian0* pingnij adres 8.8.8.8; zaobserwuj, że otrzymywana jest odpowiedź. Pingnij ten sam adres z maszyny *Virbian2*. Czy zapytanie dochodzi do maszyny *Virbian0*? Czy otrzymywana jest odpowiedź? Dlaczego? Zastanów się, kogo należałoby powiadomić o konfiguracji naszej sieci, żeby odpowiedź z komputera 8.8.8.8 wracała do maszyny *Virbian2*.

Z V0 możemy pingać 8.8.8.8, bo jesteśmy połączeni z internetem.

Z V2 ten ping dochodzi do V0, ale nie ma odpowiedzi (sprawdzimy Wiresharkiem). Należałyby powiadomić router, aby odpowiedź wracająca do maszyny V2 (patrn. uzupełn.)

## Niepunktowane zadanie dodatkowe

Poniższe zadanie można wykonać na zajęciach lub po ich zakończeniu. Warto je wykonać przed programowaniem pierwszego zadania programistycznego (*traceroute*).

- Uruchom maszynę wirtualną *Virbian0* z domyślną konfiguracją sieciową (NAT). Następnie zmień nazwę jej karty sieciowej na `enp0` i uzyskaj konfigurację sieciową poleciem

```
V0#> dhclient -v enp0
```

- Obejrzyj przypisany w ten sposób adres IP i tablice routingu poleceniami `ip addr` i `ip route`. Który wpis jest wpisem bramy domyślnej? Czy wiesz, jakie urządzenie jest tą bramą?
- Wykonaj polecenie `traceroute` do jakiegoś znanego Ci adresu IP (np. 8.8.8.8) lub nazwy domeny (np. [wikipedia.com](https://wikipedia.com)) wykorzystując opcję `-I` (jest to wariant programu `traceroute` wykorzystujący pakiety *ICMP echo request*).

```
V0#> traceroute -I 8.8.8.8
```

- Pobierz ze strony wykładu program `icmp_receive.c`.<sup>2</sup> Skompiluj go i uruchom poleceniemi

```
V0$> gcc -W -Wall -O2 icmp_receive.c -o icmp_receive  
V0#> ./icmp_receive
```

Wykonaj w maszynie wirtualnej polecenie `ping 8.8.8.8`. Zaobserwuj i porównaj pakiety odbierane w Wiresharku i programie `icmp_receive`. Powtórz eksperiment wykorzystując polecenie `traceroute -I 8.8.8.8`.

Materiały do kursu znajdują się w systemie Canvas: <https://canvas.ii.uni.wroc.pl/>.

*Marcin Bieńkowski*

---

<sup>2</sup>Pobieranie z sieci nie jest oczywiście jedynym sposobem przesyłania plików do Virbiana. Wybierając w ustawieniach maszyny część *Shared folders* można łatwo udostępnić Virbianowi dowolny katalog nadzawanego systemu operacyjnego.