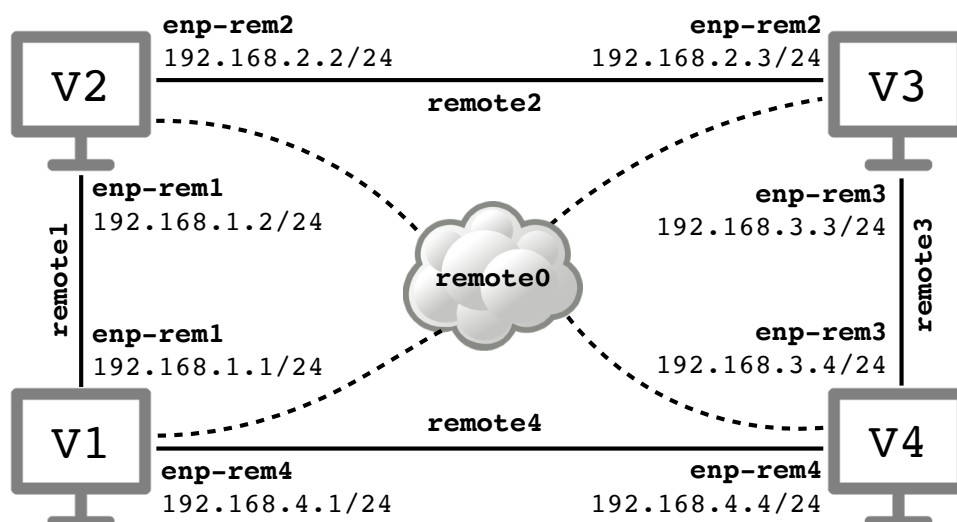


# Warsztaty z Sieci komputerowych

## Lista 4

### Konfiguracja początkowa

Celem tej części jest osiągnięcie topologii sieci jak na rysunku poniżej.

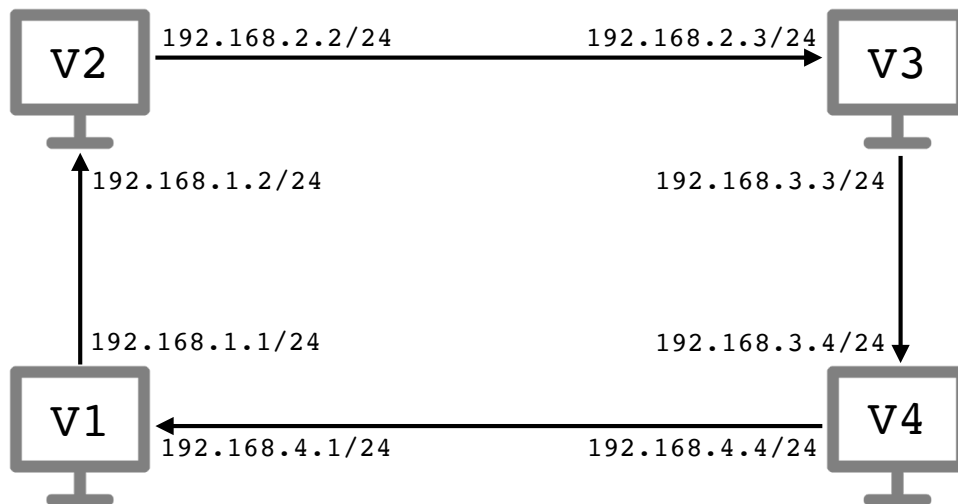


- ▶ Na każdej z czterech maszyn wirtualnych *Virbian1–Virbian4* powinny być dwa interfejsy połączone z odpowiednimi wirtualnymi sieciami *remote1–remote4*. Interfejs połączony z siecią *remotei* należy w maszynie wirtualnej nazwać *enp-remi* tak jak na rysunku powyżej. Dodatkowo na każdej maszynie powinien być interfejs nazwany *enp-all* (niezaznaczony rysunku) połączony (linie przerywane) z wirtualną siecią *remote0*.
- ▶ Na każdej maszynie aktywuj dwa interfejsy sieciowe *enp-remi*; interfejsy *enp-all* pozostaw nieaktywne. Aktywnym interfejsom przypisz adresy IP jak na rysunku powyżej. Zauważ, że karty podpięte do sieci *remotei* mają adresy IP z klasy 192.168.*i*.0/24.
- ▶ Poleceniem `ip route` sprawdź, że tablica routingu każdej maszyny zawiera dokładnie dwa wpisy dotyczące bezpośrednio połączonych z nią sieci. Sprawdź dostępność bezpośrednio połączonych maszyn poleceniem `ping`.

## Tutorial #1

Uruchom Wiresharka na wszystkich maszynach nasłuchującego na wszystkich interfejsach.

- Będziemy teraz przekazywać wszystkie pakiety do celu zgodnie ze wskazówkami zegara. Jako bramę domyślną dla każdej maszyny ustaw maszynę, która jest następna w cyklu (tj. tak jak pokazują strzałki na rysunku poniżej). Pamiętaj, że bramą powinna być osiągalna bezpośrednio karta sieciowa: przykładowo bramą domyślną dla komputera *Virbian2* powinna być równa 192.168.2.3 a nie 192.168.3.3. Upewnij się, że tablica routingu każdej maszyny zawiera dokładnie trzy wpisy.



- Poleceniem `ping` sprawdź, że z każdej maszyny osiągalne są wszystkie interfejsy innych maszyn. Prześledź w Wiresharku ścieżki komunikatów *ICMP echo request* i *ICMP echo reply*. Czy zawsze suma tych ścieżek daje pełny cykl? Dlaczego?
- Z maszyny wirtualnej *Virbian1* wykonaj polecenie `traceroute` do adresów IP przypisanych interfejsom innych maszyn. Wykorzystaj opcję `-n`, aby przyspieszyć działanie programu (wyłącza ono odpytywanie DNS).

Zauważ, że jeśli TTL pakietu wysyłanego przez `traceroute` kończy się na maszynie nie docelowej, która nie jest bezpośrednio połączona z *Virbian1*, to wyświetlany jest adres interfejsu, który *wysyła* odpowiedź ICMP na próbny pakiet, a nie adres interfejsu, który *otrzymuje* próbny pakiet. W rozważanym przypadku cykl jest na tyle mały, że taki przypadek zachodzi tylko jeśli z maszyny *Virbian1* wykonujemy `traceroute` do adresu 192.168.3.4.
- Usuń trasy domyślne z tablic routingu. Sprawdź, że zmiany odniosły skutek wyświetlając bieżącą tablicę poleceniem `ip route`.

## Tutorial #2

Skonfigurujemy teraz tablice routingu za pomocą protokołu routingu dynamicznego OSPF.

- Na każdej maszynie w pliku `/etc/frr/daemons` zmień wiersz zawierający `ospfd=no` na `ospfd=yes`. Następnie uruchom usługę `frr` poleceniem

```
Vi#> systemctl start frr
```

Aktywność usługi routingu dynamicznego RIP możesz sprawdzić poleceniem `systemctl status frr`: w wyświetlanych komunikatach powinien znajdować się napis `ospfd state -> up`.

- ▶ Na każdej maszynie uruchom konsolę `vttysh`. W razie potrzeby przypomnij sobie, jak z niej korzystać na podstawie listy zadań z poprzednich warsztatów. W konsoli `vttysh` wyświetl bieżącą tablicę routingu poleceniem

```
virbian# show ip route
```

Wyświetlane powinny być trasy do dwóch bezpośrednio podłączonych sieci `enp-remi`.

- ▶ Wejdź w tryb konfiguracji routingu OSPF poleceniami

```
virbian# configure terminal
virbian(config)# router ospf
```

Następnie włącz protokół OSPF dla sieci przyłączonych do interfejsów `enp-remi`. W tym celu wykonaj polecenia

```
virbian(config-router)# network 192.168.x.0/24 area 0
virbian(config-router)# network 192.168.y.0/24 area 0
```

Jeśli pomylisz się przy wpisywaniu, sieć można usunąć poleceniem

```
virbian(config-router)# no network adres_sieci area 0
```

- ▶ Wyjdź z trybu konfiguracji i wyświetl aktualną konfigurację poleceniami

```
virbian(config-router)# end
virbian# show running-config
```

Upewnij się, że są w niej informacje takie jak

```
router ospf
 network 192.168.x.0/24 area 0.0.0.0
 network 192.168.y.0/24 area 0.0.0.0
```

a następnie zapisz bieżącą konfigurację poleceniem

```
virbian# copy running-config startup-config
```

- ▶ Obejrzyj w Wiresharku przesyłane pakiety protokołu OSPF. Czy protokół OSPF korzysta z warstwy transportowej czy też jest osadzony bezpośrednio w pakietach IP?
- ▶ Okresowo wyświetlaj bieżącą tablicę routingu poleceniem `show ip route` w powłoce `vttysh` i poleceniem `ip route` w zwykłym terminalu.

- Polecenie `show ip route` w powłoce `vtysh` niestety nie wyświetla wyliczonej odległości do znanych sieci. Aby ją wyświetlić, wykonaj polecenia `show ip route 192.168.x.0/24` dla każdej z sieci `remote $x$`  i wynikach znajdź wpisy `metric`.
- Po zakończeniu budowania tablic poleceniami `ping` i `traceroute` sprawdź osiągalność interfejsów wszystkich maszyn.
- Na wszystkich maszynach poleceniem `ip` aktywuj interfejs `enp-all` i przypisz mu adres `172.16.16.x/16`, gdzie  $x \in \{1, 2, 3, 4\}$  jest numerem maszyny.
- W każdej maszynie włącz protokół OSPF również dla nowej sieci `172.16.0.0/16`. W tym celu w trybie konfiguracji narzędzia `vtysh` wykonaj polecenia

```
virbian(config)# router ospf
virbian(config-router)# network 172.16.0.0/16 area 0
```

Zaobserwuj przesyłane pakiety OSPF i zmiany w tablicy routingu. Zauważ, że w przypadku równej odległości dostępnych jest kilka tras do celu w tablicy routingu (w programie `frr`) natomiast nie ma ich w tablicy przekazywania (w programie `ip`).

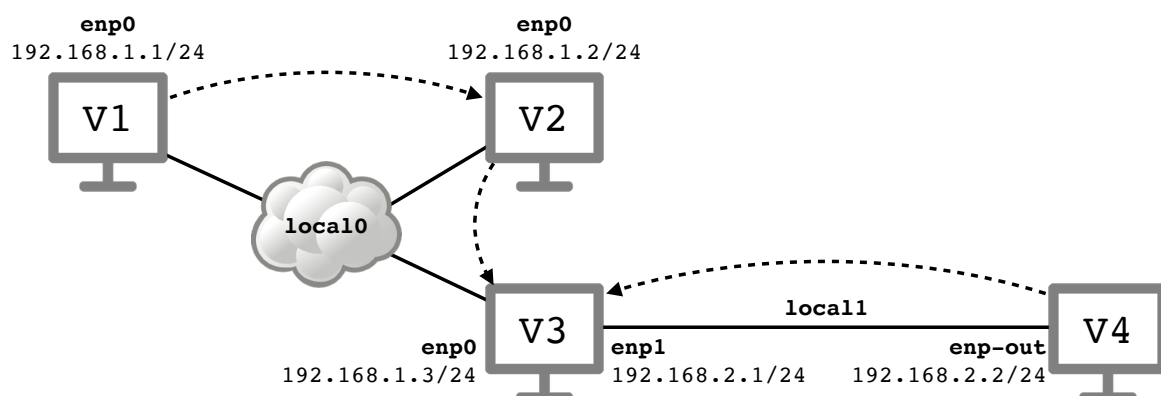
- Poleceniem `ip link set down dev interfejs` wyłącz niektóre z interfejsów, tak aby uzyskać w tablicy routingu ścieżkę o długości co najmniej 3 (wartość w polu `metric` powinna wynosić co najmniej 300). Czy umiesz uzyskać ścieżkę o długości 4?
- Zatrzymaj usługę OSPF poleceniem

```
Vi#> systemctl stop frr
```

zdekonfiguruj wszystkie interfejsy i wyłącz maszyny wirtualne.

## Wyzwanie #1

Twoim celem jest konfiguracja adresów i routingu dla topologii sieci przedstawionej na rysunku poniżej.



- 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`).

- ▶ 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.
- ▶ Poleceniem `ping` sprawdź wzajemną osiągalność maszyn podłączonych do tej samej sieci `local0` i maszyn podłączonych do tej samej sieci `local1`.
- ▶ 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`.
- ▶ Włącz Wiresharka 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?

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

*Marcin Bieńkowski*