

Warsztaty z Sieci komputerowych

Lista 4

Zadanie 1. Wszystkie komputery są wpięte do przełącznika kartami `enp3s0`. Rozpocznij zadanie od usunięcia systemowej konfiguracji tego interfejsu ale pozostawienia go aktywnym. W tym celu wydaj polecenia:

```
#> ifdown enp3s0
#> ip link set up dev enp3s0
```

Uruchom Wiresharka i włącz w nim obserwację interfejsu `enp3s0`. Jeśli otrzymujesz dużo komunikatów od przełącznika, możesz je odfiltrować klikając prawym przyciskiem myszy odpowiedni komunikat, wybierając z kontekstowego menu *prepare as filter — not selected* a następnie przycisk *apply*. Wydadź polecenie

```
#> dhclient enp3s0
```

Jakie komunikaty zostają wymienione pomiędzy Twoim komputerem a serwerem DHCP? Zauważ, że DHCP posługuje się protokołem UDP. Jaki jest w takim razie źródłowy adres IP wysyłanego pakietu, skoro Twój komputer nie ma jeszcze IP?

Obejrzyj tablicę ARP poleceniem

```
$> ip neigh
```

Usuń z niej wszystkie wpisy poleceniem

```
#> ip neigh flush all
```

Zaobserwuj, że w momencie, kiedy wywołujesz polecenie `ping adres_IP`, Twój komputer najpierw wysyła zapytanie ARP, otrzymuje na nie odpowiedź, a następnie wysyła komunikaty *ICMP echo* i otrzymuje na nie odpowiedzi. Sprawdź, że po otrzymaniu komunikatu ARP, odpowiedni wpis pojawia się w tablicy ARP.

Obejrzyj dokładnie powyższe komunikaty, zwracając uwagę na to, jakie informacje zapisane są w poszczególnych warstwach oraz na szesnastkową postać ramki. W szczególności sprawdź odpowiedzi na następujące pytania:

- Co jest danymi w przypadku zapytań ARP?
- Czy zapytania ARP są wysyłane do konkretnego komputera?
- Czy odpowiedzi ARP są wysyłane do konkretnego komputera?

Za pomocą protokołu ARP można też szybko sprawdzić jakie komputery z danej sieci IP są w lokalnym Ethernetie:

```
#> arp-scan --interface=enp3s0 adres_sieci/maska
```

Na końcu zdekonfiguruj interfejs `enp3s0` poleceniami

```
#> dhclient -r enp3s0
#> ip link set down dev enp3s0
```

Pierwsze polecenie zwalnia dzierżawiony adres IP (sprawdź to w Wiresharku), co jest miłym, choć nie absolutnie koniecznym gestem w stosunku do serwera DHCP i zatrzymuje demona `dhclient`.

Zadanie 2. Uruchom Wiresharka i włącz w nim obserwację interfejsu `enp1s0`. Interfejs `enp1s0` łączy Cię z komputerem sąsiada. Przypisz do `enp1s0` adres z sieci `192.168.1.0/24` poleceniami

```
#> ip link set up dev enp1s0
#> ip addr add adres_ip/24 dev enp1s0
```

Poleceniem `ip addr` wyświetl interfejsy sieciowe. Zwróć uwagę na przypisane im adresy IP i adresy MAC kart sieciowych. Pingnij komputer sąsiada i obejrzyj przesyłane ramki w Wiresharku. Jakie są pola nadawcy i odbiorcy ramki Ethernetowej i zawartego w niej pakietu IP?

Pingnij adres rozgłoszeniowy poleceniem

```
#> ping -b 192.168.1.255
```

i ponownie obejrzyj przesyłane ramki w Wiresharku zwracając szczególną uwagę na pola nadawcy i odbiorcy.

Na komputerze swoim lub sąsiada zmień adres IP na adres z sieci `192.168.2.0/24`:

```
#> ip addr del stary_adres_ip/24 dev enp1s0
#> ip addr add nowy_adres_ip/24 dev enp1s0
```

Zauważ, że obecnie w jednym segmencie Ethernetu są zdefiniowane dwie sieci IP. Pingnij adres rozgłoszeniowy swojej sieci i zaobserwuj przesyłane pakiety na Wiresharku uruchomionym na obu komputerach. Czy pakiet ICMP dociera również do sąsiada, pomimo tego, że jego adres IP nie należy do sieci, do której pakiet jest skierowany? Co jest wpisane w docelowym adresie ramki? Co dzieje się po tym, jak pakiet dotrze do sąsiada i dlaczego?

Na końcu zdekonfiguruj interfejs `enp1s0` poleceniami

```
#> ip addr flush dev enp1s0
#> ip link set down dev enp1s0
```

Zadanie 3. W tym zadaniu utworzymy konfigurację umożliwiającą testowanie drugiego zadania programistycznego (o routingu dynamicznym) na jednym komputerze. Stworzymy sieć złożoną z trzech komputerów, które są połączone ze sobą trzema sieciami w trójkąt.

Stwórz wirtualne interfejsy sieciowe. Każdy z nich będzie wirtualnym przełącznikiem odpowiadającym jednej sieci.

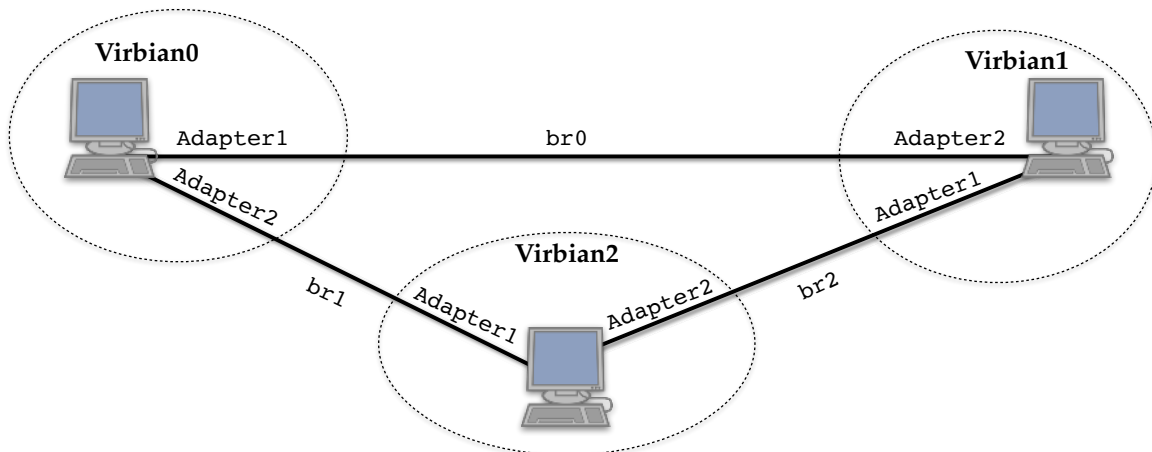
```
#> ip link add name br0 type bridge
#> ip link add name br1 type bridge
#> ip link add name br2 type bridge
#> ip link set up dev br0
#> ip link set up dev br1
#> ip link set up dev br2
```

Wyświetl włączone interfejsy sieciowe poleceniami `ip link` i `ip addr`. Interfejsy `br*` nie mają i nie będą mieć przypisanych adresów IP (mogą mieć IPv6).

Włącz teraz program VirtualBox i wybierz w nim maszynę wirtualną *Virbian0*. Po prawej stronie okna programu kliknij napis *Network* i skonfiguruj jakie interfejsy sieciowe fizycznej maszyny będą widoczne w maszynie wirtualnej. W tym celu w zakładce *Adapter 1* wybierz *Attached to: bridged adapter* a następnie z listy *Name* wybierz `br0`. W zakładce *Adapter 2* wybierz *Attached to: bridged adapter*, zaś na liście *Name* wybierz `br1`.

Do wirtualnej maszyny możesz zalogować się na konto `student` (hasło `student.8`) i na konto `root` (hasło `root.8`). Interfejs graficzny można uruchomić poleceniem `startx`. Po zalogowaniu poleceniem `ip addr` sprawdź, jakie interfejsy są dostępne w maszynie wirtualnej: powinny być dostępne dwa interfejsy `enp*s*`. Porównując wyświetlane przez `ip link` adresy MAC z opisem w zakładkach *Adapter 1* i *Adapter 2* sprawdź, który z interfejsów odpowiada *Adapter 1* a który *Adapter 2*.

Jeśli któryś z tych interfejsów ma systemowo przypisany adres IP, to należy wyłączyć tę konfigurację poleceniem `ifdown interfejs`. Interfejs odpowiadający *Adapter 1* będzie połączony z wirtualnym przełącznikiem `br0`, zaś interfejs odpowiadający *Adapter 2* będzie połączony z `br1`, jak na poniższym rysunku.



Powtórz powyższe operacje dla drugiej maszyny wirtualnej łącząc jej *Adapter 1* z interfejsem `br0` i *Adapter 2* z interfejsem `br2`, a następnie dla trzeciej łącząc jej *Adapter 1* z interfejsem `br2` i *Adapter 2* z interfejsem `br0`. Stworzoną konfigurację opisuje powyższy rysunek.

Ustal i zapisz na kartce (rozłączne adresowo) adresy IP tych trzech sieci oraz adresy IP poszczególnych kart sieciowych w maszynach wirtualnych. Skonfiguruj je w odpowiednich

maszynach wirtualnych poleceniami `ip addr`, a następnie sprawdź osiągalność połączonych interfejsów i nieosiągalność niepołączonych poleceniem `ping`.

Na końcu wyłącz wszystkie maszyny wirtualne i usuń interfejsy `br*` poleceniami

```
#> ip link del dev br0
#> ip link del dev br1
#> ip link del dev br2
```

Zadanie 4. W tym i kolejnym zadaniu zajmiemy się siecią bezprzewodową. Aktywuj interfejs `wlp5s0` poleceniem

```
#> ip link set up dev wlp5s0
```

Poleceniem

```
#> iwlist wlp5s0 scan
```

wyświetl dostępne punkty dostępowe. Widoczny powinien być m.in. punkt należący do instytutowej sieci `eduroam` oraz sieć w pracowni o identyfikatorze `lab109`. Na podstawie wyświetlanych informacji spróbuj odczytać, jakie jest szyfrowanie w poszczególnych sieciach. Jakie prędkości transmisji obsługują te sieci? Na jakich kanałach nadają te punkty?

Najpierw wykorzystamy program `iwconfig` do konfiguracji warstwy łącza danych związuąc interfejs `wlp5s0` z punktem dostępowym w pracowni. (Zauważ, że w przypadku Ethernetu warstwy drugiej nie musieliśmy konfigurować). Wyдай polecenie

```
#> iwconfig wlp5s0 essid lab109
```

Poleceniem `iwconfig` sprawdź, że co prawda identyfikator sieci został ustalony, ale nadal nie jesteśmy związani ze stacją bazową (`not associated`). Wynika to z tego, że sieć ta jest zabezpieczona kluczem WPA2, którego jeszcze nie podaliśmy.

Szyfrowanie i uwierzytelnianie WPA2 obsługuje demon `wpa_supplicant`. W dowolnym katalogu utwórz plik konfiguracyjny `wpa.conf` z następującą zawartością:

```
ctrl_interface=DIR=/var/run/wpa_supplicant
network={
    ssid="lab109"
    scan_ssid=1
    key_mgmt=WPA-PSK
    psk="bardzotajne"
}
```

Następnie uruchom program `wpa_supplicant` poleceniem

```
#> wpa_supplicant -c wpa.conf -i wlp5s0
```

Od tej pory warstwa łącza danych jest już skonfigurowana, a komputer jest związany ze stacją bazową sieci `lab109` (zweryfikuj to poleceniem `iwconfig`).

Do konfiguracji warstwy sieci wystarczy wykonać polecenie

```
#> dhclient wlp5s0
```

Obejrzyj teraz przypisany adres, tablicę routingu i sprawdź działanie sieci poleceniami `ip addr`, `ip route` i `ping`. W parach sprawdźcie również jaka jest przepustowość takiego bezprzewodowego łącza. W tym celu pierwsza osoba powinna uruchomić polecenie

```
$> iperf3 -s
```

zaś druga polecenie

```
$> iperf3 -c IP_pierwszego_komputera
```

Jak bardzo spada przepustowość, jeśli uruchamiacie to polecenie wszyscy naraz?

Poleceniem `iwlist wlp5s0 scan` sprawdź jakie są wolne kanały. Zanotuj je; przyda się to w kolejnym zadaniu.

Zauważ, że na obserwowane w Wiresharku ramki przechodzące przez interfejs `wlp5s0` wyglądają jak ramki ethernetowe. Nie jest to przypadek: sprzęt lub sterownik zamienia ramki WLAN na taką postać.

Oddaj przypisane IP poleceniem `dhclient -r wlp5s0`, zakończ program `wpa_supplicant` kombinacją `Ctrl + C` w odpowiednim terminalu lub wydając polecenie `wpa_cli terminate`. Następnie uruchom polecenie

```
#> airmon-ng start wlp5s0
```

Utworzy to wirtualny interfejs `wlp5s0mon`, który umożliwi oglądanie faktycznych ramek WLAN. Obejrzyj je w innej instancji Wiresharka i porównaj z ramkami przechodzącymi przez interfejs `wlp5s0`. Ile adresów MAC znajduje się w ramce WLAN zawierającej dane? Czemu odpowiada dodatkowy adres MAC?

Na końcu wyłącz interfejsy `wlp5s0` i `wlp5s0mon` poleceniami

```
#> airmon-ng stop wlp5s0
```

```
#> ip link set down wlp5s0
```

Zadanie 5. W tym zadaniu stworzymy sieci *ad-hoc* (tj. bez punktu dostępowego). Połączcie się w kilka grup. (Algorytmy stosowane przez sieci *ad-hoc* w Linuksie nie zawsze działają; grupy dwuosobowe powinny działać bez problemu, z większymi może być problem). Niech każda grupa wybierze sobie nazwę sieci i (najlepiej wolny) kanał.

Poleceniem `ip link` upewnij się, że interfejs `wlp5s0` jest nieaktywny (DOWN). Przełącz ten interfejs w tryb *ad-hoc* poleceniem

```
#> iwconfig wlp5s0 mode ad-hoc
```

Ustal nazwę sieci i kanał poleceniami

```
#> iwconfig wlp5s0 essid nazwa
```

```
#> iwconfig wlp5s0 freq numer
```

Uaktywnij interfejs poleceniem

```
#> ip link set up dev wlp5s0
```

Następnie — wewnątrz każdej grupy — przypisz kartom adresy IP z jednej sieci IP i sprawdź czy komputery widzą się nawzajem. Ewentualne problemy można próbować diagnozować poleceniem `iwconfig wlp5s0`: każda ze stacji w grupie powinna mieć taką samą nazwę sieci (nazwę ESSID), taką samą częstotliwość i taki sam adres komórki (*cell*). Sprawdź połączenie poleceniem `ping` i przepustowość programem `iperf3`.

Na końcu usuń konfigurację interfejsu `wlp5s0` poleceniami

```
#> ip addr flush dev wlp5s0
#> ip link set down dev wlp5s0
```

Lista i materiały znajdują się pod adresem <http://www.ii.uni.wroc.pl/~mbi/dyd/>.

Marcin Bienkowski