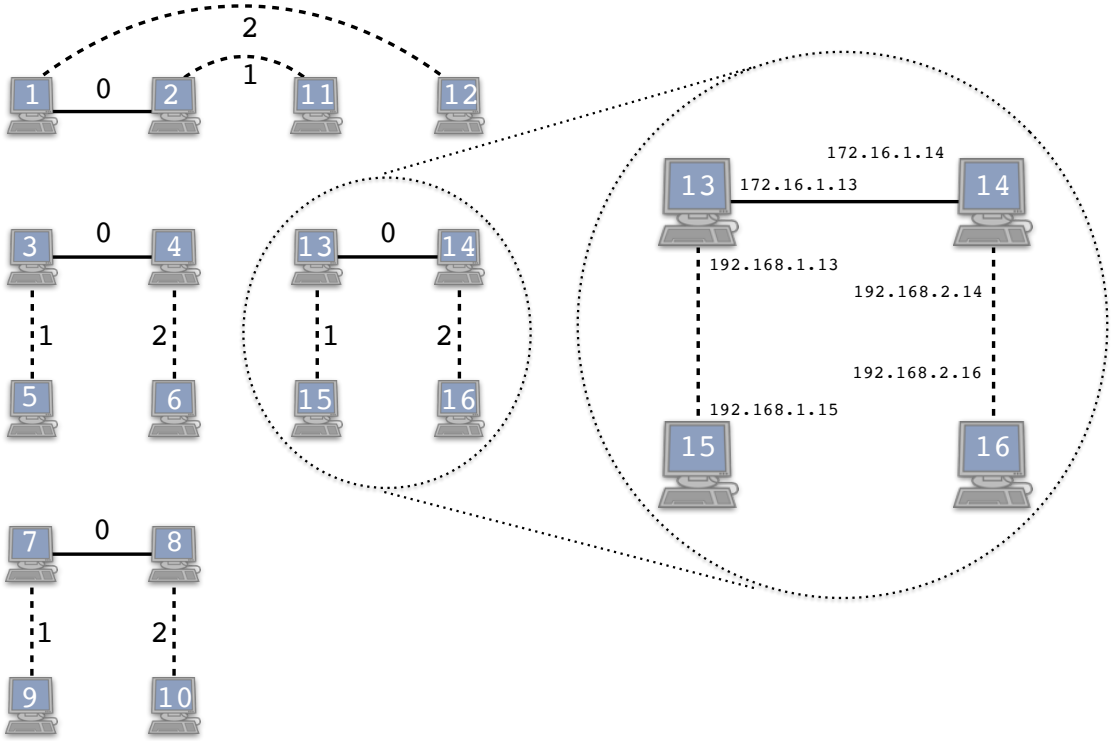
Warsztaty z Sieci komputerowych
Lista 2

1 Uwagi ogólne

Topologia sieci na te zajęcia została przedstawiona poniżej; każda czwórka komputerów jest osobną strukturą niepołączoną z niczym innym. Linią ciągłą oznaczono połączenia między interfejsami `eth0`, zaś przerywaną — między interfejsami `eth1`. Na początku pracy wydaj polecenie `sudo netmode lab`.

Sieć w pracowni składa się z niezależnych czwórek komputerów jak na rysunku poniżej. Celem tej pracowni jest konfiguracja ich interfejsów sieciowych i routingu pomiędzy sieciami (sieci są dwupunktowymi połączeniami zaznaczonymi na rysunku).

Komputery, które mają połączone karty `eth0` (komputery o numerach 1, 2, 3, 4, 7, 8, 13, 14) będziemy nazywać komputerami *typu A*, zaś pozostałe komputerami *typu B*.



2 Zadania

Zadanie 1. Wszystkim komputerom skonfiguruj warstwę sieciową interfejsu `eth1` poleceniem

```
#> ifconfig eth1 192.168.y.x netmask 255.255.255.0
```

gdzie $x \in \{1, \ldots, 16\}$ jest numerem komputera, zaś $y \in \{1, 2\}$ jest numerem krawędzi (patrz rysunek wyżej). Przykładowo karta komputera nr 8 powinna otrzymać adres `192.168.2.8`.

Następnie interfejsom `eth0` komputerów typu A przypisz adres IP równy `172.16.1.`x, gdzie x jest numerem komputera. W tym celu wydaj polecenie

```
#> ifconfig eth0 172.16.1.x netmask 255.255.0.0
```

Zwróć uwagę na inną w tym przypadku maskę podsieci. Wyświetl aktualnie skonfigurowane interfejsy poleceniami

```
#> ifconfig
#> ip addr
```

To drugie polecenie jest nowocześniejszym narzędziem zarządzającym interfejsami sieciowymi. Obejrzyj też aktualną tablicę routingu poleceniem

```
#> route -n
```

Za pomocą programu `ping` sprawdź, czy połączone bezpośrednio ze sobą komputery „widzą się" wzajemnie.

Zadanie 2. Zauważ, że masz skonfigurowany interfejs `lo`. Pingnij adres pętli lokalnej `127.0.0.1`. Zauważ, że komunikaty dochodzą, pomimo tego, że nie ma odpowiedniego wpisu w tablicy routingu. W rzeczywistości wpis takowy wewnętrznie istnieje, ale polecenie `route -n` go po prostu nie wyświetla. Aby się o tym przekonać, przeanalizuj wynik polecenia

```
#> routel
```

Włącz Wiresharka nasłuchując na wszystkich interfejsach i zaobserwuj, co jest wypisywane w konsoli oraz jakie pakiety są wysyłane i odbierane jeśli pingasz:

1. adres `127.0.0.1`;
2. swój własny adres IP przypisany do interfejsu `eth1`;
3. adres IP sąsiedniego komputera podłączonego do interfejsu `eth1`;
4. adres rozgłoszeniowy sieci podłączonej do interfejsu `eth1` (poleceniem `ping -b 192.168.1.255` lub `ping -b 192.168.2.255`);
5. nieistniejący adres IP należący do sieci podłączonej do interfejsu `eth1`;
6. adres z sieci, do której nie jesteś bezpośrednio podłączony, np. `10.10.10.10`.

Porównaj otrzymane komunikaty, przesyłane pakiety i czasy reakcji.

Zadanie 3. Z komputera typu B sprawdź osiągalność karty `eth0` jego sąsiada typu A (połączonego za pomocą karty `eth1`):

```
$> ping 172.16.1.x
```

gdzie x jest numerem komputera sąsiada. Przykładowo na komputerze nr 5 należy wydać polecenie `ping 172.16.1.3`. Adres ten jest nieosiągalny, gdyż nadawca nie wie jak dostać się do sieci `172.16.0.0/16`. Spróbujmy to naprawić dodając trasę domyślną, która przechodzi przez osiągalną bezpośrednio kartę sąsiada:

```
#> route add default gw 192.168.y.x
```

gdzie $x \in \{1, \ldots, 16\}$ jest numerem komputera typu A, zaś $y \in \{1, 2\}$ jest numerem incydentnej krawędzi (patrz rysunek). Przykładowo na komputerze nr 16 należy wydać polecenie `route add default gw 192.168.2.14`. Jeśli pomylisz się wpisując polecenie `route`, dodaną pomyłkowo trasę możesz skasować poleceniem `route del` (z identycznymi opcjami, co w przypadku polecenia `route add`). Wyświetl bieżącą tablice routingu poleceniem

```
#> route -n
```

Spróbuj teraz wykonać poprzednie polecenie `ping` (powinno zakończyć się sukcesem).

Czy oznacza to, że inne adresy z sieci `172.16.0.0/16` są osiągalne? Aby to sprawdzić, pingnij drugi adres IP z sieci `172.16.0.0/16`, tj. należący do drugiego komputera typu A. Przykładowo na komputerze nr 5 należy wydać polecenie `ping 172.16.1.4`. Zapamiętaj to polecenie; będziemy je określać mianem „pingnij najdalszy interfejs `eth0`".

Co jest przyczyną niepowodzenia? Jaki komunikat otrzymujesz? Na komputerach typu A sprawdź Wiresharkiem, że odpowiedni komunikat ICMP jest otrzymywany i przekazywany do komputera docelowego. Dlaczego więc nie jest odsyłana odpowiednia odpowiedź?

Zadanie 4. Na komputerach typu A dodaj trasę prowadzącą do sieci, która nie jest do niego bezpośrednio połączona:

```
#> route add -net 192.168.y.0/24 gw 172.16.1.x
```

gdzie x jest numerem sąsiedniego komputera A, zaś $y \in \{1, 2\}$ jest numerem krawędzi odpowiadającej tej sieci. Na komputerze typu B pingnij najdalszy interfejs `eth0`. Dlaczego ostatnie polecenie `route` pomogło w otrzymywaniu odpowiedzi na `ping`?

Zadanie 5. Przedstawiony na rysunku obraz sieci nie jest kompletny. W rzeczywistości komputery typu A należą do jednej wspólnej sieci `172.16.0.0/16` podłączonej za pośrednictwem routera `172.16.255.252` do Internetu. Na komputerach typu A skonfiguruj trasę domyślną do Internetu poleceniem

```
#> route add default gw 172.16.255.252
```

Ze wszystkich komputerów pingnij jakiś znany Ci istniejący adres IP (np. `8.8.8.8`). Obejrzyj Wiresharkiem wszystkie przesyłane komunikaty. Dlaczego ping z komputerów typu A udaje się, a z komputerów typu B nie? Kogo należałoby powiadomić o sieciach `192.168.y.x`? Jak można inaczej rozwiązać ten problem?

Zadanie 6. Na wszystkich komputerach zdekonfiguruj interfejs `eth1` poleceniem

```
#> ifconfig eth1 down
```

a na komputerach typu B dodatkowo interfejs `eth0` poleceniem

```
#> ifconfig eth0 down
```

Na wszystkich komputerach uzyskaj konfigurację interfejsu `eth0` poleceniem

```
#> ifup eth0
```

Obejrzyj przypisany w ten sposób adres IP poleceniem `ifconfig`. Teraz wszystkie komputery są połączone interfejsem `eth0` z siecią `172.16.0.0/16` i za pośrednictwem routera `172.16.255.252` z resztą Internetu. (Fizyczne połączenie zawsze istniało, ale można było je do tej pory zignorować, gdyż interfejs `eth0` komputerów typu B był nieaktywny).

Wykonaj polecenie `traceroute` do jakiegoś znanego Ci adresu IP (np. `8.8.8.8`) lub nazwy domeny (np. `wikipedia.com`). Zaobserwuj przesyłane pakiety Wiresharkiem.

Uruchom wirtualną maszynę (program VirtualBox, maszyna Debian Wheezy 0, użytkownik i hasło dostępne są w opisie maszyny). Po zalogowaniu do wirtualnej maszyny ponownie wypróbuj polecenie `traceroute` oraz jego wariant wykorzystujący pakiety *ICMP echo request*:

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

Ten ostatni wariant wymaga uprawnień administratora. W obu przypadkach Wiresharkiem obejrzyj przesyłane pakiety. W razie potrzeby odfiltruj wszystkie pakiety poza tymi, które są skierowane do Twojego adresu IP, lub z niego wychodzą wpisując w Wiresharku filtr `ip.addr == Twój_adres_IP`. (Filtr `ip.addr == a.b.c.d` jest równoważny filtrowi `ip.src == a.b.c.d || ip.dst == a.b.c.d`).

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

Marcin Bieńkowski