

Wstęp do systemu operacyjnego UNIX

Laboratorium 9:

Podstawy protokołu TCP/IP w systemie Linux

W sieci ethernet urządzenia są rozróżnialne według adresów zwanych adresami ethernetowymi lub sprzętowymi (hardwarowymi, MAC – angMedia Access Control). Adresy te zostały wpisane do karty sieciowej, a ich niepowtarzalność jest zapewniona podziałem puli dostępnych adresów między producentów sprzętu. Adresy te składają się z 6-ciu bajtów (48 bitów) i służą nawiązywaniu połączenia na poziomie sprzętowym. Teoretycznie nie ma możliwości zmiany adresu ethernetowego.

Postać adresu internetowego jest określona przez specyfikację protokołu IP (Interconnection Protocol). W chwili obecnej najczęściej używaną wersją jest wersja IV. Tworzone są obecnie „wyspy”, w których obowiązuje wersja VI, jednak pełne wprowadzenie tej wersji obliczone jest na kilka następnych lat.

W wersji IV protokołu IP adres składa się z 4 bajtów (32 bitów) i podobnie jak w przypadku adresu ethernetowego zakłada się, że nie istnieją w sieci internet dwa urządzenia o takim samym adresie. Jak łatwo obliczyć, możliwości adresowe protokołu są niewielkie, gdyż po odliczeniu pewnych adresów zarezerwowanych mamy do dyspozycji ok 2.1×10^6 niepowtarzalnych adresów. Dodatkowo, wprowadzony sztywny podział na tzw. klasy adresowe dodatkowo ograniczył rozmiar tej puli. Stąd też wprowadzono tzw. metodę routingu bezklasowego, która umożliwiła podział puli adresowej bardziej elastycznie, „prawie” zgodnie z potrzebami klientów. Metoda ta wymaga podania tzw. maski podsieci określającej liczbę dostępnych w danej podsieci adresów (np. 255.255.255.192 oznacza $256-192=64$ dostępne adresy, a $255.255.252.0 - (256-252)*256=1024$ dostępne adresy).

Każdy komputer w sieci lokalnej swój adres IP może mieć na stałe wpisany w odpowiednich plikach konfiguracyjnych lub adres ten może uzyskiwać od wybranego komputera w sieci. Ten wyróżniony komputer pełni funkcję serwera adresów, a protokół uzyskiwania adresu IP nazywa się DHCP (Dynamic Host Configuration Protocol).

Przyjęto, że dwa komputery w podsieci mogą połączyć się bezpośrednio, jeżeli ich adresy są zgodne w części odpowiedzialnej za numer sieci. W przypadku, gdy pakiet informacji jest adresowany do urządzenia, którego adres różni się na więcej niż ostatnim bajcie jest on kierowany do urządzenia zwanego routerem lub bramą (gateway), które to urządzenie dzięki zapisanej w nim liście adresów (tablicy routingu) wie gdzie pakiet przesłać dalej.

Wielokrotnie, podczas pracy wielu programów sieciowych pojawia się konieczność wysłania zapytania-pakietu do wszystkich komputerów w sieci lokalnej. Wykorzystywany jest wówczas adres rozgłoszeniowy zwany broadcastem. Pakiety adresowane tym adresem nie przechodzą przez routery - „żyją” jedynie w granicach sieci lokalnej.

Oprócz adresu IP komputery posiadają również adresy w postaci symbolicznej. Adres taki składa się z nazwy komputera (do pierwszej kropki od lewej strony) oraz nazwy domeny. Np. thorin.icsr.agh.edu.pl oznacza adres komputera o nazwie thorin, z domeną icsr.agh.edu.pl. Przejście pomiędzy adresem IP i symbolicznym może zostać zapisane w pliku konfiguracyjnym na każdym komputerze sieci (sieć płaska) lub też na wybranych komputerach sieci, które w razie konieczności będą odpytywane przez inne komputery w sieci o odpowiednie adresy. Usługa ta nazywa się DNS (Domain Name Service).

W chwili obecnej głównym punktem zainteresowań są zagadnienia związane z adresem IP. W systemach Unix, podstawową komendą związaną z zagadnieniami sieciowymi jest komenda ifconfig.

0. Podłącz się do systemu jako użytkownik root.

1. Korzystając z manuala zapoznaj się z opcjami oraz argumentami komendy ifconfig.

Podstawowe informacje o konfiguracji wszystkich interfejsów sieciowych uzyskamy uruchamiając komendę ifconfig z opcją -a.

```
1 [root@thorin root]# ifconfig -a
2 eth0      Link encap:Ethernet  HWaddr 00:50:04:DF:3D:4B
3           inet addr:192.168.3.116  Bcast:192.168.3.127  Mask:255.255.255.128
4           UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
5           RX packets:0 errors:0 dropped:0 overruns:0 frame:0
6           TX packets:4 errors:0 dropped:0 overruns:0 carrier:4
7           collisions:0 txqueuelen:100
8           RX bytes:0 (0.0 b)  TX bytes:240 (240.0 b)
```

```

9      Interrupt:11 Base address:0x4000
10
11 lo      Link encap:Local Loopback
12          inet addr:127.0.0.1  Mask:255.0.0.0
13          UP LOOPBACK RUNNING  MTU:16436  Metric:1
14          RX packets:10 errors:0 dropped:0 overruns:0 frame:0
15          TX packets:10 errors:0 dropped:0 overruns:0 carrier:0
16          collisions:0 txqueuelen:0
17          RX bytes:700 (700.0 b)  TX bytes:700 (700.0 b)

```

W lewej kolumnie pojawia się nazwa interfejsu. W rozpatrywanym systemie zainstalowana jest jedna karta sieciowa typu ethernet, a w systemie jest rozpoznawana jako `eth0`. Jej adres sprzętowy to: `00:50:04:DF:3D:4B`, IP: `192.168.3.116`, adres rozgłoszeniowy: `192.168.3.127` oraz maska podsieci: `255.255.255.128`. Kolejne informacje to statystyka protokołu. Informacje zamykają dane o konfiguracji sprzętowej - numer przerwania oraz adres bazowy urządzenia w pamięci.

Interfejs `lo` to pętla zwrotna (`loopback`). Pakiety wysłane na adres `127.0.0.1` trafiają po tym interfejsie do komputera, który je wysłał.

2. Korzystając z manuala zapoznaj się z podstawowymi opcjami i argumentami komendy `ping`.

3. Korzystając z komendy `ping` sprawdź czy komputer o numerze 127.0.0.1 odpowiada na pakiety ping. Sprawdź czy komunikacja zostanie przerwana poprzez odłączenie kabla sieciowego.

Komenda `ifconfig` może również służyć do konfigurowania interfejsów. Nadanie adresu IP oraz maski podsieci dla interfejsu `eth0` oraz aktywowanie go może wyglądać następująco:

```

1 [root@thorin root]# ifconfig eth0 192.168.3.116 netmask 255.255.255.128 broadcast 192.168.3.127 up

```

4. Korzystając z komendy `ifconfig` nadaj swojemu komputerowi odpowiedni adres IP oraz maskę sieciową dla interfejsu `eth0`. Uczyń interfejs aktywnym. O adresy zapytaj prowadzącego.

W systemie **RedHat Linux**, w katalogu `/etc/sysconfig/network-scripts` tworzone są pliki o nazwach `ifcfg-XXXX`, gdzie XXXX oznacza nazwę danego interfejsu. Przykładowo, interfejsowi `eth0` odpowiada plik `ifcfg-eth0`. Są to pliki tekstowe. Jeśli adres IP jest nadany statycznie, to plik `/etc/sysconfig/network-scripts/ifcfg-eth0` będzie wyglądał jak poniżej:

```

1 DEVICE=eth0
2 BOOTPROTO=static
3 BROADCAST=192.168.3.127
4 IPADDR=192.168.3.116
5 NETMASK=255.255.255.128
6 NETWORK=192.168.3.0
7 ONBOOT=yes

```

Jeśli system otrzymuje adres z serwera DHCP plik ten może wyglądać następująco:

```

1 DEVICE=eth0
2 BOOTPROTO=dhcp
3 ONBOOT=yes

```

5. Uzupełnij wpisy w pliku `/etc/sysconfig/network-scripts/ifcfg-eth0`, tak aby twój komputer posiadał odpowiedni adres IP, informację o sposobie uzyskiwania (`static`) i adresie rozgłoszeniowym. O brakujące adresy zapytaj prowadzącego.

W chwili obecnej, do podłączenia do sieci globalnej brakuje jeszcze adresu bramy. Mówiąc bardziej ogólnie konieczne jest podanie drogi (`route`) pakietu do urządzenia będącego bramą w naszej sieci lokalnej. Służy do tego komenda `route` . Przykładowo, jeśli bramą w naszej podsieci ma być urządzenie o adresie `192.168.3.1` postać komendy będzie następująca:

```
1 [root@thorin root]# route add default gw 192.168.3.1 eth0
```

6. Zapoznaj się ze składnią komendy `route` . Zdefiniuj w systemie domniemaną bramę sieci o adresie uzyskanym od prowadzącego.

7. Sprawdź komendą `ping` , czy urządzenie zdefiniowane jako domniemana brama jest dostępne po interfejsie `eth0` .

Wprowadzone zmiany mają charakter tymczasowy. Aby były aktywne również po przeładowaniu systemu operacyjnego należy je wprowadzić do pliku `/etc/sysconfig/network` . W tej dystrybucji systemu plik ten powinien wyglądać:

```
1 NETWORKING=yes
2 HOSTNAME=thorin.icsr.agh.edu.pl
3 GATEWAY=192.168.3.1
```

Innym sposobem jest ustawianie adresu bramy, oraz nazwy komputera każdorazowo podczas ładowania systemu. Komendę `route add` należy wówczas umieścić w pliku `/etc/rc.d/rc.local` .

8. Uzupełnij zawartość pliku konfiguracyjnego `/etc/sysconfig/network` . Zrestartuj komputer (komenda `shutdown`) i podłącz się ponownie jako użytkownik `root` .

Tablica routingu zawiera informację o możliwych trasach pakietów z naszego systemu do sieci. Pełną informację udostępnia komenda `netstat` z opcjami `-nr` :

```
1 [root@thorin network-scripts]# netstat -nr
2 Kernel IP routing table
3 Destination      Gateway          Genmask         Flags   MSS Window  irtt Iface
4 192.168.3.0       0.0.0.0         255.255.255.128 U        40 0        0 eth0
5 127.0.0.0         0.0.0.0         255.0.0.0       U        40 0        0 lo
6 0.0.0.0           192.168.3.1     0.0.0.0         UG       40 0        0 eth0
```

Przykładowo przez interfejs `eth0` - pierwsza linia listingu - możemy dostać się do sieci `192.168.3.0` z wykorzystaniem bramy `0.0.0.0` (sieć podłączone jest bezpośrednio do tego interfejsu), sieć ma maskę `255.255.255.128` i interfejs ten jest aktywny. Ostatnia linia listingu dotyczy połączenia z siecią globalną (`0.0.0.0`) przy pomocy urządzenia `192.168.3.1` .

9. Sprawdź w manulu komendy `netstat` znaczenie opcji `n` oraz `r` . Jak wygląda tablica routingu w Twoim systemie?

Informacja o „przejściu” między adresem symbolicznym, a adresem IP dla komputera pracującego pod kontrolą systemu operacyjnego UNIX jest zapisana w pliku `/etc/hosts` . Jest to plik tekstowy, w którym jedna linia definiuje jeden komputer. Plik ten zawiera również wpis o pętli zwrotnej. Dawniej, gdy usługa DNS nie była tak rozpowszechniona i w pliku tym umieszczane były informacje o wszystkich „uczęszczanych” komputerach. W chwili obecnej stracił on nieco na znaczeniu, niemniej jednak w wielu wypadkach jest pierwszym do którego odwołuje się wiele aplikacji dla znalezienia odpowiedniego adresu. Może on wyglądać następująco:

```
1 # Do not remove the following line, or various programs
2 # that require network functionality will fail.
3 127.0.0.1          localhost.localdomain localhost
4 192.168.3.116      thorin.icsr.agh.edu.pl
```

Każda linia pliku zawiera kolejno: adres IP, adres symboliczny, oraz dodatkowo może zawierać zero lub więcej nazw krótkich (przezwoisk lub aliasów) dla danego komputera.

10. Uzupełnij zawartość pliku `/etc/hosts` o nazwę komputera kolegi obok. Sprawdź czy komenda `ping` akcetuje dopisaną nazwę oraz aliasy.

11. Zapoznaj się z funkcjami oraz ze składnią komendy `hostname`. Użyj komendy `hostname` do sprawdzenia nazwy systemu, w którym pracujesz.

Plik `/etc/hosts` pozwala na rozpoznawanie nazw symbolicznych wybranych komputerów. Co zatem z pozostałymi? Tu wykorzystamy usługę DNS. Klient tej usługi (komputer poszukujący zdresu IP na podstawie znajomości adresu symbolicznego) wykorzystuje swój lokalny plik `/etc/resolv.conf` do odnalezienia adresu IP serwerów usługi DNS oraz nazwy domeny, w której dany klient pracuje. Plik ten, jest plikiem tekstowym i może wyglądać następująco:

```
1 search filemon.agh.edu.pl
2 nameserver 192.168.3.10
3 nameserver 149.156.96.9
```

Każda linia pliku rozpoczyna się od słowa kluczowego, a po nim oddzielone co najmniej jednym białym znakiem pojawiają się odpowiednie wartości oddzielone przecinkami. Słowem kluczowym może być:

- `nameserver` – zawiera adres IP serwera DNS. Zalecane jest podanie co najmniej dwóch serwerów. W tym przypadku specyfikujemy je po jednym w linii. Wykorzystywane są w kolejności w jakiej występują na liście.
- `domain` – nazwa domeny, w której dany klient pracuje. Np. dla komputera `thorin.icsr.agh.edu.pl` nazwa domeny to `icsr.agh.edu.pl`.
- `search` – nazwy domen (oddzielone spacjami), które mają być odpytywane w przypadku próby rozwinięcia nazwy symbolicznej. Do podanej przez użytkownika krótkiej (jednoczłonowej) nazwy nazwy „doklejane” są nazwy przeszukiwanych domen i dopiero takie pytanie jest kierowane do serwera DNS.

12. Uzupełnij zawartość pliku `/etc/resolv.conf` podając adres pierwszego serwera DNS: `192.168.3.10`, drugiego: `217.96.89.130`. Nazwa domeny to: `filemon.agh.edu.pl`. Zrestartuj system. Podłącz się ponownie jako użytkownik `root`.

13. W tym momencie nazwy symboliczne powinny być poprawnie rozwiązywane. Sprawdź czas wędrówki pakietu do komputera `www.uci.agh.edu.pl`. Jaki jest adres IP tego komputera?

Oprócz czasu wędrówki pakietu istotna jest również jego trasa. Śledzenie trasy pakietu między naszym systemem, a innym urządzeniem w sieci jest możliwe dzięki komendzie `traceroute`. Poniżej zamieszczono przykład jej wykorzystania:

```
1 [root@thorin root]# traceroute -I artemis.wszib.edu.pl
2 traceroute to artemis.wszib.edu.pl (217.96.89.130), 30 hops max, 40 byte packets
3  1 ucitr.agh.edu.pl (149.156.96.4)  1.395 ms  1.754 ms  1.403 ms
4  2 manrtr.cyf-kr.edu.pl (149.156.6.217)  0.944 ms  0.833 ms  0.780 ms
5  3 router.wszib.krakow.pl (149.156.234.2)  3.158 ms  3.082 ms  3.345 ms
6  4 artemis.wszib.edu.pl (217.96.89.130)  697.779 ms  3.973 ms  3.685 ms
```

Nagłówek listingu informuje nas o nazwie symbolicznej i adresie IP komputera docelowego, maksymalnej liczbie urządzeń aktywnych sieci przez które przejdzie pakiet zanim zostanie usunięty oraz o rozmiarze pakietu. Kolejne linie to nazwy i adresy IP urządzeń - o ile zechcą one te dane udostępnić oraz czasy wędrówki.

14. Korzystając z manuala komendy `traceroute` sprawdź, co oznaczają kolejne czasy w liniach odpowiadających kolejnym urządzeniom aktywnym. Sprawdź trasę pakietu do komputera `luke.icsr.agh.edu.pl`.

Z protokołów `TCP` lub `UDP` może w tym samym czasie korzystać więcej niż jeden proces użytkownika. Musi zatem istnieć metoda rozróżniania danych należących do poszczególnych procesów. Do identyfikowania danych konkretnego procesu obydwa protokoły używają 16-bitowych liczb całkowitych zwanych numerami portów.

Plik `services`¹ plikiem tekstowym, który umożliwia przyporządkowanie zrozumiałych nazw usług do odpowiednich numerów portów i rodzajów protokołów (patrz dokumentacja wbudowana – `service(5)`). Każda aplikacja sieciowa powinna konsultować z plikiem `/etc/services` numer portu (i protokołu) dla określonej usługi – patrz funkcja `getservbyname(3)`.

Każdy wiersz tego pliku opisuje jedną usługę i ma następujący format:

service-name port/protocol [aliases ...]

gdzie:

- `service-name` – nazwa usługi. Rozróżnia się małe i wielkie litery. Często program- klient jest nazywany tak jak `service-name`.
- `port` – numer portu (liczba dziesiętna) jaki ma być wykorzystywany.
- `protocol` – rodzaj protokołu jaki ma zostać użyty. To pole powinno zawierać jeden z protokołów zdefiniowanych w pliku `protocols(5)`. Zwykle jest to `tcp` lub `udp`.
- `aliases` – opcjonalna lista innych nazw tej samej usługi (rozdzielonych spacjami lub znakami tabulacji).

Poszczególne pola mogą być rozdzielane spacjami lub znakami tabulacji.

Komentarze rozpoczynają się od znaku `#` aż do końca wiersza. Puste wiersze są pomijane.

```
1 # Each line describes one service, and is of the form:
2 #
3 # service-name port/protocol [aliases ...] [# comment]
4
5 tcpmux      1/tcp      # TCP port service multiplexer
6 tcpmux      1/udp      # TCP port service multiplexer
7 rje         5/tcp      # Remote Job Entry
8 rje         5/udp      # Remote Job Entry
9 .....
10 # 21 is registered to ftp, but also used by fsp
11 ftp         21/tcp
12 ftp         21/udp      fsp fspd
13 ssh         22/tcp      # SSH Remote Login Protocol
14 ssh         22/udp      # SSH Remote Login Protocol
15 telnet      23/tcp
16 telnet      23/udp
17 # 24 - private mail system
18 smtp        25/tcp      mail
19 smtp        25/udp      mail
20 time        37/tcp      timserver
21 time        37/udp      timserver
22 .....
23 finger      79/tcp
24 finger      79/udp
25 .....
```

Obecność pozycji w pliku `services` nie oznacza, że dana usługa jest obsługiwana przez system, oznacza jedynie, że system zna nazwę i numre portu.

Do obsługi wielu usług internetowych jest wykorzystywany demon `inetd` nazywany demonem Internet, jego konfiguracja jest zapisana w pliku `inetd.conf(5)`. Obecne dystrybucje posiadają zazwyczaj nowszą wersję tego demona, który nazywa się `xinetd` (*ang. eXtended internet service daemon*). Zadanie rozszerzonego demona jest identyczne a konfiguracja podobna. Cenną cechą nowego demona jest możliwość rozdzielania dużego i nieporęcznego pliku konfiguracyjnego znanego z wersji poprzedniej na małe pliki odpowiedzialne za poszczególne usługi, zmiana taka ułatwia automatyczne dodawanie/usuwanie usług podczas instalacji usług.

Rozpatrzmy przykład serwera `finger`, którego zadaniem jest udostępnienie informacji o użytkownikach pracujących w systemie. Serwer ten jest uruchamiany poprzez demon internet, stosowny dla tej usługi plik konfiguracyjny to `/etc/xinetd.d/finger`. Jego zawartość jest następująca:

¹którego istnienie zostało wymuszone poprzez normy również w innych systemach operacyjnych.

```

1 service finger
2 {
3     socket_type    = stream
4     wait           = no
5     user           = nobody
6     server         = /usr/sbin/in.fingerd
7     disable        = yes
8 }

```

Gdzie w pierwszej linii po słowie kluczowym występuje nazwa usługi, czyli nazwa pochodząca z pliku `/etc/services` – w tym wypadku jest to usługa `finger`. Kolejne linie ujęte w blok poprzez nawiasy klamrowe są atrybutami dotyczącymi sposobu uruchomienia konkretnej usługi. Ważniejsze atrybuty to:

socket_type – typ połączenia (usługi) w zasadzie spotkamy się tu jedynie z wartościami `dgram` czyli gniazdo przesyłające poszczególne datagramy oraz `stream` czyli ciągły strumień danych. W praktyce dla protokołu IP `stream` oznacza połączenie poprzez `tcp` zaś `dgram` oznacza `udp`,

wait – możliwe wartości to `yes` lub `no`, pole to informuje demon internet czy należy oczekiwać na zakończenie działania programu przed obsługą kolejnego żądania.

user – nazwa użytkownika w koncie którego uruchomiony zostanie proces obsługujący usługę,

server – ścieżka do programu który zostanie uruchomiony,

disable – wartość `yes` lub `no` czy usługa ma być obsługiwany czy też nie.

Należy pamiętać o tym że nie wszystkie usługi sieciowe świadczone przez komputer są uruchamiane poprzez `inetd`. W szczególności serwery informacyjne (`NNTP`) i poczty (`SMTP`) czy `WWW` są zwykle uruchamiane pośrednio ze skryptów startujących (uruchamianych przez `init(8)`).

15. Zmień konfigurację demona `xinetd` tak aby obsługiwał usługę `finger`. Aby zachęcić działającego demona `xinetd` do ponownego odczytania pliku/plików konfiguracyjnych można wyłączyć do niego stosowny sygnał (patrz dokumentacja `xinetd`).

16. Sprawdź dostępność usługi z sąsiedniego komputera.

Polecenie `netstat`, którym już posługiwaliśmy się, pozwala wyświetlić znacznie więcej informacji na temat konfiguracji sieci. Polecenie wydane bez argumentów wyświetla listę nawiązanych połączeń. Z argumentem `-a` wyświetla informacje na temat wszystkich połączeń (również tych które nie są jeszcze nawiązane a oczekują na połączenie).

17. Sprawdź jakie usługi są uruchomione na twoim komputerze.

System plików `/proc` stanowi reprezentację tabeli aktywnych procesów jądra oraz konfiguracji jądra systemu. Operując na plikach i katalogach w katalogu `/proc` można poznać i zmienić różnego rodzaju parametry działającego systemu. Należy pamiętać, że manipulowanie na katalogu `/proc` jako użytkownik `root` może być niebezpieczne, gdyż jedna błędna operacja może np. spowodować zniszczenie całej pamięci jądra. Działając na zawartości katalogu `/proc` spróbujemy zwiększyć wydajność protokołu TCP/IP.

W pierwszym kroku uczynimy aktywnymi kilka składników protokołu TCP. Uczynimy to wpisując jedyńki w odpowiednich plikach:

```

1 echo 1 > /proc/sys/net/ipv4/tcp_timestamps
2 echo 1 > /proc/sys/net/ipv4/tcp_window_scaling
3 echo 1 > /proc/sys/net/ipv4/tcp_sack

```

18. Wprowadź zmiany w plikach `tcp_timestamps`, `tcp_tcp_window_scaling` oraz `tcp_sack` jak pokazano powyżej. Jakie jest znaczenie wprowadzonych zmian?

Kolejne zmiany będą dotyczyć rozmiaru odbieranych i wysyłanych ramek oraz rozmiarów buforów komunikacyjnych przechowujących pakiety wysyłane i odbierane. Wielkości te definiują zawartości następujących plików:

- `/proc/sys/net/core/rmem_default` - rozmiar domniemany odbieranej ramki,
- `/proc/sys/net/core/rmem_max` - rozmiar maksymalny odbieranej ramki,
- `/proc/sys/net/core/wmem_default` - rozmiar domniemany wysyłanej ramki,
- `/proc/sys/net/core/wmem_max` - rozmiar maksymalny wysyłanej ramki,
- `/proc/sys/net/ipv4/tcp_rmem` - obszar pamięci zarezerwowany dla bufora pakietów odbieranych,
- `/proc/sys/net/ipv4/tcp_wmem` - obszar pamięci zarezerwowany dla bufora pakietów wysyłanych,

Zawartość plików ustawiamy według wzoru:

```

1 echo 8388608 > /proc/sys/net/core/wmem_max
2 echo 8388608 > /proc/sys/net/core/rmem_max
3 echo "4096 87380 4194304" > /proc/sys/net/ipv4/tcp_rmem
4 echo "4096 65536 4194304" > /proc/sys/net/ipv4/tcp_wmem

```

19. Wprowadź zmiany według podanego powyżej schematu.

20. Sprawdź w manualu, do czego służą i jaka jest składnia komend `ifdown` i `ifup`. Zamknij interfejs `eth0`.