helion.pl
KSIĘGARNIA INTERNETOWA · WYDAWNICTWO

IDŹ DO
PRZYKŁADOWY ROZDZIAŁ
SPIS TREŚCI

KATALOG KSIĄŻEK
KATALOG ONLINE
ZAMÓW DRUKOWANY KATALOG

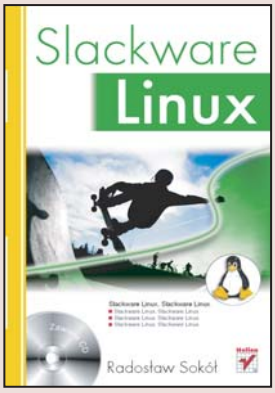
TWÓJ KOSZYK
DODAJ DO KOSZYKA

CENNIK I INFORMACJE
ZAMÓW INFORMACJE O NOWOŚCIACH
ZAMÓW CENNIK

CZYTELNIA
FRAGMENTY KSIĄŻEK ONLINE

Slackware Linux

Autor: Radosław Sokół
ISBN: 83-246-0055-8
Format: B5, stron: 304



Uruchom własny serwer sieciowy, korzystając z Linuksa

- Instalacja i konfiguracja systemu
- Administrowanie Linuksem
- Uruchamianie usług sieciowych

Coraz większe możliwości dostępu do internetu oferowane przez dostawców usług telekomunikacyjnych otwierają nowe horyzonty nie tylko przed firmami, ale również przed użytkownikami domowymi. Dziś praktycznie każdy użytkownik komputera może podłączyć swój sprzęt do internetu za pomocą łącza stałego. Posiadając takie łącze, można pokusić się o samodzielne zbudowanie sieci – obejmującej kilka mieszkań lub tylko jedno. Do tego niezbędny będzie jednak serwer „obsługujący" dostęp do sieci. Linux – dostępny nieodpłatnie system operacyjny, tworzony i rozwijany przez pasjonatów z całego świata – jest doskonałą bazą dla takiego serwera. Skromne wymagania sprzętowe pozwalają uruchomić go nawet na starym komputerze, który trudno już wykorzystać do innych celów.

Czytając książkę „Slackware Linux", poznasz jedną z najpopularniejszych „odmian" Linuksa, zwanych dystrybucjami: Slackware. Dowiesz się, skąd można pobrać ten system operacyjny i jak zainstalować go na komputerze. Nauczysz się konfigurować go i zarządzać jego zasobami. Poznasz metody uruchamiania usług niezbędnych do poprawnego działania sieci, zabezpieczania sieci przed atakami hakerów i wirusami, a także monitorowania obciążenia łącza. Przeczytasz także o samodzielnym kompilowaniu jądra Linuksa, pozwalającym dostosować system do potrzeb użytkownika.

- Pobieranie obrazów płyt instalacyjnych z internetu
- Instalacja Linuksa
- Uruchamianie i zatrzymywanie systemu
- Praca z powłoką tekstową
- Zarządzanie systemem plików i procesami
- Administrowanie kontami użytkowników
- Instalowanie nowych aplikacji
- Współdzielenie łącza sieciowego
- Konfiguracja zapory sieciowej
- Uruchamianie serwera DHCP, DNS, FTP i HTTP
- Poczta elektroniczna
- Statystyki działania serwera

Poznaj alternatywny system operacyjny i wykorzystaj go w domowej sieci komputerowej

Wydawnictwo Helion
ul. Kościuszki 1c
44-100 Gliwice
tel. 032 230 98 63
e-mail: helion@helion.pl

Spis treści

Wstęp **9**

Rozdział 1. Instalacja systemu **17**

Przygotowanie do instalacji systemu 18
Strona WWW projektu Slackware Linux 18
Pobieranie obrazów płyt instalacyjnych 18
Instalacja systemu 21
Rozpoczynanie instalacji systemu 21
Systemy plików 22
Podział dysku twardego na partycje 26
Uruchamianie instalatora 30
Wybór pakietów 36
Dalsza instalacja systemu 39
Konfiguracja myszy 44
Konfiguracja połączenia sieciowego 45
Ostateczna konfiguracja systemu 45
Pierwsze uruchomienie systemu 49

Rozdział 2. Zarządzanie serwerem **51**

Logowanie i wylogowywanie się 51
Zamykanie i ponowne uruchamianie systemu 52
Kończenie pracy systemu 53
Ponowne uruchamianie systemu 53
Konsole wirtualne 54
Wbudowana dokumentacja systemu 55
Hierarchiczna struktura katalogów 57
Podstawowe polecenia powłoki systemu 58
Automatyczne uzupełnianie nazw poleceń, katalogów i plików 59
Obsługa klawiatury i ekranu 60
Zarządzanie katalogami 60
Zarządzanie plikami 64
Uprawnienia do plików i katalogów 69
Przetwarzanie potokowe 70
Łączenie wielu poleceń 73
Tworzenie skryptów 73

Archiwizowanie plików ... 76
Zarządzanie procesami ... 78
Wyświetlanie listy działających procesów ... 78
Awaryjne kończenie pracy procesów ... 80
Uruchamianie programu ze zmienionym priorytetem ... 82
Program top ... 82
Konta użytkowników ... 85
Uzyskiwanie uprawnień administratora ... 86
Wyświetlanie listy kont i grup użytkowników ... 86
Zakładanie kont użytkowników ... 88
Zmiana hasła ... 89
Usuwanie konta użytkownika ... 90
Użytkownicy a grupy ... 90
Korzystanie z nośników optycznych ... 91
Zarządzanie pakietami oprogramowania ... 92
Instalacja pakietu ... 92
Aktualizacja pakietu ... 93
Narzędzie pkgtool ... 93
Narzędzie swaret ... 94
Midnight Commander ... 98
Uruchamianie programu ... 99
Poruszanie się po katalogach ... 99
Operacje ... 100
Skróty klawiszowe ... 101

Rozdział 3. Sieci ... 103
TCP/IP w teorii ... 103
Adresy IP ... 106
Pule adresów publicznych i niepublicznych ... 107
Połączenia TCP ... 108
Interfejsy sieciowe ... 109
Ramki danych ... 110
Adresy sprzętowe ... 110
Wyświetlanie listy interfejsów sieciowych ... 111
Aktywacja interfejsu sieciowego ... 112
Konfiguracja połączenia sieciowego ... 112
Ustalanie domyślnej konfiguracji sieci ... 113
Wprowadzanie adresu serwera DNS ... 114
Wyświetlanie adresów IP przypisanych interfejsom ... 115
Dynamiczna konfiguracja interfejsów sieciowych ... 115
Diagnostyka łączy sieciowych ... 116
ping ... 116
traceroute ... 117
netstat ... 118
iptraf ... 118

Rozdział 4. Dzielenie łącza sieciowego ... 123
Współdzielenie łącza sieciowego ... 123
Aktywacja dzielenia łącza ... 124
Konfiguracja komputerów-klientów ... 126
Testowanie działania współdzielenia łącza sieciowego ... 127
Przekierowywanie portów ... 129
Bufor stron WWW Squid ... 130
Instalacja pakietu Squid ... 131
Konfiguracja pakietu Squid ... 133

Uruchamianie i zatrzymywanie serwera Squid ... 135
Testowanie działania serwera pośredniczącego Squid ... 136
Dzienniki zdarzeń pakietu Squid ... 137
Przezroczysty serwer pośredniczący ... 139

Rozdział 5. Zabezpieczanie serwera ... 141
Możliwe ataki i zagrożenia ... 141
Aktualizacja oprogramowania ... 143
Tworzenie haseł ... 143
Przeglądanie dziennika zdarzeń ... 146
Konfiguracja i zabezpieczanie zdalnego dostępu do serwera ... 147
Uzyskiwanie zdalnego dostępu do serwera ... 148
Konfiguracja usługi SSH ... 149
Blokada możliwości „zgadywania" haseł ... 151
Zapora sieciowa iptables ... 152
Wprowadzenie ... 152
Potrzeba stosowania zapór sieciowych ... 153
Podstawy działania zapory iptables ... 154
Konfiguracja zapory iptables ... 157
Opcje tworzące definicję reguły ... 162
Przykłady konfiguracji zapory iptables ... 163
Automatyczne tworzenie kopii zapasowych ... 169
Tworzenie kopii zapasowej katalogu ... 170
Regularne automatyczne tworzenie kopii ... 170
Odtwarzanie kopii zapasowej danych ... 173

Rozdział 6. Serwer DHCP ... 175
Przygotowanie serwera DHCP do pracy ... 176
Sprawdzenie obecności oprogramowania serwera DHCP ... 176
Instalacja oprogramowania serwera DHCP ... 176
Aktualizacja oprogramowania serwera DHCP ... 176
Uruchamianie i zatrzymywanie serwera DHCP ... 177
Konfiguracja serwera DHCP ... 179
Opcje globalne ... 179
Deklaracja zakresu adresów IP ... 180
Określanie niezmiennych adresów IP ... 181
Przykładowy plik konfiguracyjny ... 182
Ponowne uruchamianie serwera DHCP ... 183
Testowanie działania serwera DHCP ... 183
Monitorowanie działania serwera DHCP ... 184

Rozdział 7. Serwer DNS ... 187
Przygotowanie serwera BIND do pracy ... 189
Sprawdzenie obecności pakietu BIND ... 190
Instalacja pakietu BIND ... 190
Aktualizacja pakietu BIND ... 190
Uruchamianie i zatrzymywanie serwera DNS ... 191
Konfiguracja serwera DNS ... 192
Sprawdzanie poprawności konfiguracji ... 192
Opcje konfiguracyjne ... 193
Przykładowa sekcja options ... 193
Ustalanie serwera DNS jako domyślnego dla systemu ... 194
Testowanie działania serwera ... 194

Domeny ... 195
Strefa prosta ... 196
Strefa odwrotna ... 200
Diagnostyka serwera DNS ... 202
Polecenie host ... 203

Rozdział 8. Serwer HTTP ... 205
Przygotowanie serwera Apache do pracy ... 206
Sprawdzenie obecności pakietu Apache ... 206
Instalacja pakietu Apache ... 206
Aktualizacja pakietu Apache ... 208
Uruchamianie i zatrzymywanie serwera HTTP ... 209
Testowanie funkcjonowania serwera ... 210
Uaktywnianie i testowanie funkcjonowania interpretera PHP ... 210
Konfiguracja serwera Apache ... 214
Tworzenie i zabezpieczanie własnego serwisu WWW ... 218

Rozdział 9. Serwer FTP ... 221
Przygotowywanie serwera FTP do pracy ... 222
Sprawdzenie obecności oprogramowania vsftpd ... 222
Instalacja pakietu vsftpd ... 222
Aktualizacja pakietu vsftpd ... 223
Uruchamianie i zatrzymywanie serwera FTP ... 224
Konfiguracja serwera FTP ... 225
Opcje konfiguracyjne ... 226
Przykładowy plik konfiguracyjny ... 229
Testowanie działania serwera FTP ... 229
Monitorowanie działania serwera FTP ... 230

Rozdział 10. Samba ... 233
Przygotowanie pakietu Samba do pracy ... 234
Sprawdzenie obecności pakietu Samba ... 235
Instalacja pakietu Samba ... 235
Aktualizacja pakietu Samba ... 236
Uruchamianie i zatrzymywanie serwera Samba ... 237
Konfiguracja udostępniania plików ... 238
Plik konfiguracyjny ... 238
Budowa pliku konfiguracyjnego ... 239
Sekcja [global] ... 239
Sekcja [homes] ... 240
Tworzenie własnych zasobów sieciowych ... 241
Konta użytkowników ... 242
Dodawanie konta użytkownika ... 243
Zmiana hasła użytkownika ... 243
Usuwanie konta użytkownika ... 244
Korzystanie z zasobów sieciowych ... 244

Rozdział 11. Statystyki ... 249
Instalacja ... 250
Instalacja biblioteki GD ... 250
Instalacja pakietu MRTG ... 252
Testowanie poprawności działania pakietu ... 253

Konfiguracja ... 253
Tworzenie pliku konfiguracyjnego ... 253
Regularne uruchamianie pakietu MRTG ... 255
Definicja źródła danych ... 257
Tryby pomiaru danych ... 261
Zmiana częstotliwości próbkowania ... 262
Skrypty pobierające dane ... 263
Analiza obciążenia procesora ... 263
Analiza wykorzystania powierzchni dyskowej ... 263
Analiza wykorzystania łącza sieciowego ... 264
Analiza łączności sieciowej ... 265
Badanie obciążenia łącza internetowego przez poszczególnych użytkowników ... 266
Testowanie i diagnostyka ... 269

Dodatek A Słowniczek terminów i pojęć ... 271

Dodatek B Przedrostki i jednostki miary stosowane w informatyce ... 285

Dodatek C Numery portów TCP i UDP ... 289

Dodatek D Przeliczanie systemów liczbowych ... 293

Skorowidz ... 295

Rozdział 3.
Sieci

Mechanizmem, dzięki któremu prawie każde urządzenie sieciowe na świecie — niezależnie od zastosowanego w nim procesora czy systemu operacyjnego — może porozumiewać się z innymi urządzeniami podłączonymi do globalnej Sieci (w tym także z komputerami), jest protokół TCP/IP. Protokół TCP/IP (ang. *Transmission Control Protocol/Internet Protocol*), opracowany w latach 70. ubiegłego stulecia przez Vintona Cerfa oraz Boba Kahna, zaadaptowany został na potrzeby Internetu (wtedy jeszcze noszącego nazwę ARPAnet) w 1983 roku. Początkowo protokół ten miał być stosowany jedynie tymczasowo, trwały bowiem badania teoretyczne mające stać się podstawą opracowania doskonalszego protokołu. Okazało się jednak, że TCP/IP doskonale sprawdza się w realnych zastosowaniach, a jego szybkie rozpowszechnienie uniemożliwiło zastąpienie go nowszym protokołem.



Stanowi to problem również dziś. Wdrażanie nowszej wersji protokołu TCP/IP, nazwanej IPv6 i zapewniającej o wiele szerszą pulę dostępnych adresów i usług, jest utrudnione przez powszechność używanej obecnie postaci TCP/IP (nazywanej IPv4).

Nowatorską cechą protokołu TCP/IP było oparcie go o rozwijającą się dopiero technikę przełączania pakietów (ang. *packet switching*), podczas gdy w powszechnym użyciu były łącza komutowane. Przełączanie pakietów umożliwia wyeliminowanie stałego zestawiania łącza między nadawcą i odbiorcą wiadomości — to zwiększa efektywność wykorzystania łączy i pozwala przesyłać dane różnymi trasami, co przydaje się w przypadku uszkodzenia jednej z linii transmisyjnych.

TCP/IP w teorii

Nazwa TCP/IP jest nieco myląca, gdyż opisywany nią protokół ma strukturę hierarchiczną, na szczycie której znajduje się protokół IP (ang. *Internet Protocol*) dzielący się (między innymi) na następujące protokoły podrzędne:

- TCP (ang. *Transmission Control Protocol*) — protokół umożliwiający zestawianie dwukierunkowego łącza wirtualnego „punkt-punkt" (ang. *point-to--point connection*) między nadawcą i odbiorcą, a także kontrolowanie stanu tego połączenia oraz weryfikowanie poprawności przesyłania danych;
- UDP (ang. *User Datagram Protocol*) — protokół bezpołączeniowy, umożliwiający przesyłanie pojedynczych pakietów danych (tak zwanych datagramów) bez gwarancji poprawności dostarczenia oraz zachowania kolejności odbioru zgodnej z kolejnością wysyłania;
- ICMP (ang. *Internet Control Message Protocol*) — protokół bezpołączeniowy, pełniący rolę kontrolną i diagnostyczną (diagnozowanie stanu połączenia z wybranym węzłem sieci, zmiana tras przesyłania pakietów, śledzenie trasy przesyłania pakietu).

Każdy węzeł sieci IP scharakteryzowany jest unikatowym identyfikatorem liczbowym, tak zwanym adresem IP. W obecnej wersji protokołu IP, jaką jest IPv4, adres IP jest 32-bitową liczbą całkowitą, co pozwala teoretycznie podłączyć do jednej sieci ponad cztery miliardy urządzeń sieciowych. W rzeczywistości na liczbę dostępnych adresów wpływa wiele czynników:

- Fragmenty przestrzeni adresowej IP zarezerwowane są na użytek sieci lokalnych, dzięki czemu na świecie może istnieć nieograniczona liczba komputerów posiadających ten sam adres IP — pod warunkiem oczywiście, że adres ten należy do zarezerwowanej, lokalnej puli adresowej; adresy z tej lokalnej puli nie mogą pojawiać się w ogólnoświatowej sieci Internet.
- Niektóre, wyposażone w kilka kart sieciowych komputery, wykorzystują kilka adresów IP ; w efekcie liczba adresów dostępnych dla innych użytkowników końcowych jest zmniejszona.
- Wiele organizacji rezerwuje całkiem obszerne pule publicznych adresów IP, korzystając tylko z niektórych adresów z tej puli; pozostałe, wolne adresy są niedostępne dla innych firm i organizacji.



Protokół IPv6 stosuje adresy o długości 128 bitów, co pozwala niewyobrażalnie zwiększyć liczbę możliwych węzłów sieci. Mówiąc bardziej obrazowo, protokół IPv6 pozwoli przesyłać dane między czterema miliardami planet, wśród których na każdej znajdują się cztery miliardy państw, w nich — cztery miliardy sieci komputerowych, a w każdej z nich — cztery miliardy komputerów.

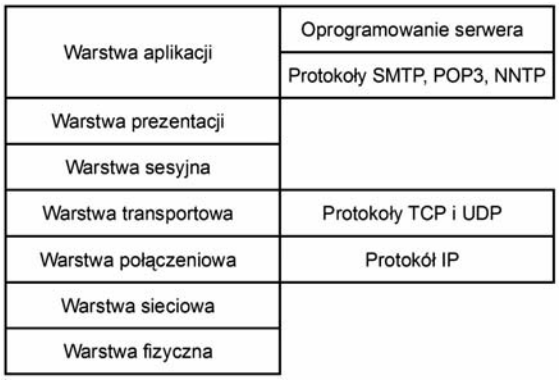
Olbrzymia większość usług internetowych — a wśród nich te najbardziej narażone na ataki z zewnątrz — działa w oparciu o protokół połączeniowy TCP. Termin *połączeniowy* określa zdolność protokołu TCP do tworzenia dwukierunkowych łączy wirtualnych „punkt-punkt" między wybranymi dwoma węzłami sieci. Warstwa TCP gwarantuje spójność przesyłu informacji zarówno pod względem jej poprawności (weryfikacja sum kontrolnych), jak i składni (pakiety danych odbierane są zawsze w kolejności identycznej z kolejnością nadania) oraz kompletności (zagubione lub zniszczone pakiety są transmitowane ponownie). Kontroluje ona również stan łącza wirtualnego, informując oprogramowanie realizujące transmisję o ewentualnym zerwaniu łącza.



Protokoły bezpołączeniowe — na przykład UDP i ICMP — przesyłają zawsze pojedyncze pakiety danych, nie troszcząc się o wcześniejsze nawiązanie połączenia, układanie przesyłanych pakietów w logiczny ciąg i sprawdzanie, czy wysłane dane zostały w ogóle odebrane.

Jedynym znaczącym efektem prac teoretycznych mających na celu opracowanie doskonalszych od TCP/IP protokołów pocztowych, który znalazł zastosowanie, był model odniesienia ISO/OSI (ang. *International Standards Organization, Open Systems Interconnection Reference Model*), opracowany w 1983 roku przez międzynarodową organizację standaryzacyjną ISO. Model ten przedstawia łącze komunikacyjne dowolnego typu w sposób abstrakcyjny, dzieląc je na siedem warstw, z których każda może odwoływać się wyłącznie do usług warstwy niższego rzędu (rysunek 3.1):

1. **Warstwa fizyczna** odpowiada za przesył danych za pośrednictwem wybranego medium transmisyjnego — kabla koncentrycznego, skrętki, światłowodu, fal radiowych.
2. **Warstwa połączeniowa** odpowiada za jednoczesny dostęp wielu urządzeń do medium transmisyjnego obsługiwanego przez warstwę fizyczną i umożliwia warstwie wyższej odbieranie i nadawanie pojedynczych fragmentów (pakietów) danych.
3. **Warstwa sieciowa** odpowiada za wybór pakietów danych przeznaczonych dla konkretnego węzła sieci oraz za ewentualne przekazywanie (tak zwane routowanie, ang. *routing*) pakietów między różnymi sieciami.
4. **Warstwa transportowa** dzieli odbierane z warstwy wyższej dane na pakiety obsługiwane przez warstwę sieciową oraz łączy pakiety odbierane z warstwy sieciowej w jednolity strumień danych, przekazywany warstwie wyższej.
5. **Warstwa sesyjna** implementuje mechanizm łączy wirtualnych, ukrywając przed warstwami wyższymi pakietowy charakter przesyłu danych.
6. **Warstwa prezentacji** przekształca dane w sposób umożliwiający wymianę danych między węzłami sieci posługującymi się odmienną reprezentacją danych.
7. **Warstwa aplikacji** jest warstwą najwyższą i obejmuje wszystkie możliwe aplikacje połączenia sieciowego.



Rysunek 3.1. *Warstwowy model ISO/OSI*



W praktyce często dwie najwyższe warstwy modelu ISO/OSI — prezentacji i aplikacji — przedstawia się jako jedną wspólną, gdyż realizowane przez nie funkcje przeplatają się nawzajem, tworząc jedną, logiczną całość.

Adresy IP

Każde wirtualne łącze TCP identyfikowane jest jednoznacznie czterema parametrami:

- adresem IP komputera nawiązującego połączenie,
- numerem portu TCP komputera nawiązującego połączenie,
- adresem IP komputera odbierającego połączenie,
- numerem portu TCP komputera odbierającego połączenie.

Adres IP to trzydziestodwubitowa liczba całkowita z zakresu od 0 do 4 294 967 296. Dla wygody zapisuje się ją w postaci czterech liczb 8-bitowych, które mogą przyjmować wartości od 0 do 255. Na przykład adres 85.14.98.13 ma następującą postać binarną i liczbową:

85	14	98	13
01010101	00001110	01100010	00001101
1 427 005 965			

Każdy adres IP dzieli się na dwie części. Pierwszą z nich jest adres *sieci IP*, do której należy komputer, a drugą *identyfikator komputera*, unikatowy w ramach tej sieci. Do podziału służy *maska sieci*, będąca również liczbą, interpretowaną jednak wyłącznie binarnie. Aby dokonać podziału, należy zapisać adres IP w formie binarnej, a zaraz pod nim — bit pod bitem — maskę sieci; tam, gdzie w masce znajduje się cyfra 1, bit adresu należy do adresu sieci, a tam, gdzie cyfra 0 — do identyfikatora komputera.

Najprostsze maski mają postać 255.0.0.0, 255.255.0.0 lub 255.255.255.0 (w zapisie z podziałem na cztery liczby dziesiętne). Liczba 255 odpowiada wszystkim ośmiu bitom ustawionym w stan 1, a to oznacza, że dany fragment adresu w całości należy do adresu sieci. Na przykład:

Adres IP i maska	Adres sieci	Identyfikator komputera
85. 14. 98. 13 255. 0. 0. 0	85.0.0.0	0.14.98.13
85. 14. 98. 13 255.255. 0. 0	85.14.0.0	0.0.98.13
85. 14. 98. 13 255.255.255. 0	85.14.98.0	0.0.0.13

Trudniej jest, gdy maska składa się z innych liczb. W takim przypadku konieczne jest niestety rozpisanie adresu IP i maski w formie binarnej i dokonanie rozdziału, na przykład tak:

Adres IP i maska (dziesiętnie)	Adres IP i maska (binarnie)	Adres sieci	Identyfikator komputera
85. 14. 98. 13	01010101 00001110 01100010 00001101	85.14.96.0	0.0.2.13
255.255.224. 0	11111111 11111111 11100000 00000000		



Pamiętaj, że maska sieci po rozpisaniu w formie binarnej musi być wyraźnie podzielona na dwie części: lewą z samymi jedynkami i prawą z samymi zerami. Jeżeli po konwersji gdzieś w masce pojawia Ci się samotna jedynka lub samotne zero ze złej strony, najprawdopodobniej pomyliłeś się w obliczeniach.



Tabelę konwersji liczb dziesiętnych na binarne i szesnastkowe znajdziesz w Dodatku D. Funkcję przeliczania liczb do innych systemów posiadają też kalkulatory (w tym komputerowe, na przykład Kalkulator systemu Windows).

Numer portu TCP to szesnastobitowa dodatnia liczba całkowita , identyfikująca jeden z 65 536 portów komunikacyjnych TCP. Istnienie wielu portów komunikacyjnych umożliwia nawiązanie wielu połączeń wirtualnych między tymi samymi dwoma węzłami sieci. Z każdym portem komunikacyjnym skojarzona może być dowolna liczba połączeń wchodzących (odebranych), ale tylko jedno połączenie wychodzące.

Pule adresów publicznych i niepublicznych

Wszystkich unikatowych adresów IP teoretycznie mogą być ponad cztery miliardy, co w czasie projektowania protokołu IP miało wystarczyć na całe dekady. Adresów IP zaczyna jednak brakować, a łatwo sobie wyobrazić, co by się działo, gdyby dwie osoby zaczęły naraz używać tego samego adresu. Poza tym względy techniczne nakazują, by cała podsieć IP (wyznaczana przez maskę sieci, o której przed chwilą czytałeś) była używana w jednym miejscu, przez jedną instytucję.

Założono zatem organizacje, których zadaniem jest przydzielanie całych zbiorów (pul) adresów IP na poziomie świata i poszczególnych państw. Każda firma lub osoba potrzebująca nowej puli adresów IP zwraca się do takiej organizacji i otrzymuje w posiadanie — za opłatą — podsieć IP zawierającą kilka, kilkanaście lub kilkadziesiąt adresów, wedle potrzeb. Najczęściej, aby uniknąć częstych formalności, pule adresowe zamawia się z naddatkiem, dzięki czemu po zakupie nowych komputerów nie trzeba kupować kolejnej puli adresowej.

Trudno jednak wymagać, aby absolutnie każdy zgłaszał korzystanie z sieci jakiejś organizacji i otrzymywał pulę adresów. Oczywiście sieci nie połączone ze sobą (przede wszystkim — nie połączone z Internetem) mogą teoretycznie używać dowolnych

adresów, gdyż w niczym to nie szkodzi i nikt tego nie sprawdzi. Problem pojawia się jednak w momencie dołączenia sieci do Internetu: adresy komputerów w sieci lokalnej nie mogą pokrywać się z adresami już używanymi na świecie.

Na ratunek idą tak zwane **pule niepubliczne** adresów IP. Są to pule przeznaczone wyłącznie do użycia w sieciach lokalnych, unikatowe jedynie w zakresie jednej sieci. Istnieją trzy takie pule:

- 10.0.0.0/8 (maska 255.0.0.0) — stosowana dosyć rzadko i odpowiednia tylko w wielkich sieciach lokalnych, zawierających setki oddziałów wyposażonych w tysiące komputerów;
- 172.16.0.0/12 (maska 255.240.0.0) — stosowana niezwykle rzadko i odpowiednia dla średnich sieci lokalnych, zawierających setki tysięcy komputerów;
- 192.168.0.0/16 (maska 255.255.0.0) — stosowana najczęściej, może obsłużyć ponad 65 000 komputerów w jednej sieci lokalnej (z dowolnym podziałem, na przykład 256 sieci po 254 komputery lub jedna sieć zawierająca 65 534 komputery — zależy to od zastosowanej maski podsieci).

Adresy należące do powyższych pul są nierozpoznawalne w Internecie. Innymi słowy, komputer z nadanym takim adresem jest niedostępny z poziomu innych komputerów podłączonych do Internetu. Również korzystanie z zasobów serwerów internetowych wymaga w przypadku komputera z niepublicznym adresem IP stosowania specjalnych środków: serwerów pośredniczących lub współdzielenia łącza internetowego technikami znanymi pod nazwami NAT (ang. *Network Address Translation*) lub, rzadziej, *maskarada*.

Posiadacz niepublicznego adresu IP może zatem w nieco ograniczonym stopniu korzystać z zasobów Internetu, nie może jednak swoich zasobów udostępniać światu. Jedynym sposobem obejścia tego ograniczenia jest *przekierowywanie portów* wykonywane na serwerze (*routerze*) posiadającym publiczny adres IP.



O współdzieleniu łącza internetowego w systemie Slackware Linux oraz metodzie realizacji przekierowywania portów możesz przeczytać w kolejnym rozdziale.

Połączenia TCP

Procedura nawiązania połączenia wirtualnego TCP przedstawia się następująco:

1. Węzeł sieci usiłujący nawiązać połączenie (na polecenie jednego z wykonywanych przez niego programów) rezerwuje jeden port komunikacyjny TCP na potrzeby połączenia i wysyła do docelowego węzła sieci pakiet danych zawierający: informację o chęci nawiązania połączenia, swój adres IP, numer zarezerwowanego portu, adres IP docelowego węzła sieci oraz numer docelowego portu TCP.
2. Docelowy węzeł sieci interpretuje otrzymany pakiet danych i wysyła informację o przyjęciu lub odrzuceniu połączenia scharakteryzowanego podanymi w poprzednim punkcie danymi.

3. Węzeł źródłowy odbiera odpowiedź. Jeśli była ona negatywna, program usiłujący nawiązać połączenie jest powiadamiany o problemie. Jeśli była pozytywna, do nadawcy wysyłany jest pakiet danych potwierdzający przyjęcie informacji o zezwoleniu nawiązania połączenia.
4. Kolejne pakiety danych przesyłane w obu kierunkach, zaadresowane tak samo jak pakiet nawiązujący połączenie (adresy IP oraz numery portów TCP charakteryzują połączenie wirtualne) służą do przesyłania właściwych danych.

Zerwanie połączenia wirtualnego może nastąpić ze strony dowolnego z dwóch węzłów, pomiędzy którymi nawiązane zostało połączenie. Strona zrywająca połączenie przesyła przeciwległemu węzłowi pakiet zawierający informację o zakończeniu połączenia. Jeśli jest to niemożliwe (na przykład wskutek fizycznego odcięcia jednego z węzłów od sieci), każda ze stron automatycznie zerwie połączenie z powodu jego przedawnienia.

Interfejsy sieciowe

Urządzenie, za pomocą którego komputer podłączany jest do sieci teleinformatycznej, nosi nazwę *interfejsu sieciowego*. Interfejsami sieciowymi są na przykład:

- karta sieciowa (na przykład karta Ethernet),
- analogowy modem telefoniczny,
- modem DSL,
- terminal ISDN.

Komputer może być wyposażony w wiele interfejsów sieciowych. Zazwyczaj zwykłe komputery użytkowników oraz serwery obsługujące tylko sieć lokalną wyposażane są w jeden interfejs sieciowy (w olbrzymiej większości przypadków: kartę sieciową Ethernet). Serwery (*routery*) łączące sieć lokalną z Internetem mają zazwyczaj dwa interfejsy sieciowe, z których co najmniej jeden będzie kartą sieciową (drugi może być zrealizowany w inny sposób, w zależności od typu posiadanego łącza internetowego). Komputery pełniące rolę dużych routerów łączących ze sobą wiele sieci mogą być wyposażone w kilka kart sieciowych z kilkoma osobnymi połączeniami sieciowymi na każdej, co w sumie może dać kilkanaście lub kilkadziesiąt interfejsów sieciowych.

W systemie Linux każdemu interfejsowi sieciowemu odpowiada nazwa składająca się z kodowego oznaczenia typu połączenia oraz kolejnego numeru. Na przykład karty sieciowe Ethernet mogą otrzymać oznaczenia `eth0` czy `eth7`, a połączenia typu PPP realizowane za pomocą modemu — `ppp3` czy `ppp5`. Szczególną nazwą interfejsu jest `lo` — odpowiada ona połączeniu zwrotnemu (ang. *loop-back*) nie reprezentowanemu przez żadne urządzenie. Połączenie zwrotne możesz traktować jako miniaturową sieć składającą się tylko i wyłącznie z Twojego komputera i niemożliwą do rozbudowy.



Interfejs sieciowy `lo` ma zawsze przydzielony adres IP 127.0.0.1/8.

Ramki danych

Większość interfejsów sieciowych przesyła dane w postaci bloków o konkretnym rozmiarze. Jednocześnie na rozmiar takiego bloku nakładany jest limit zapobiegający długotrwałemu wykorzystaniu łącza przez jeden blok danych. Ma to szczególne znaczenie w przypadku łącz sieciowych, do których równolegle podłączonych jest wiele urządzeń, a żadnemu nie wolno zmonopolizować transmisji — przykładem jest sieć lokalna Ethernet.

Maksymalny rozmiar bloku danych nosi nazwę MTU (ang. *Maximum Transmission Unit*) i określa się go w bajtach. Przykładowo: w sieci Ethernet rozmiar taki wynosi 1 500 bajtów, co oznacza, że w jednym bloku można przesłać co najwyżej 1 500 bajtów użytecznych danych (a jeżeli odliczy się dane „tracone" na nagłówki pakietów IP, efektywna użyteczna pojemność zmniejsza się o kilkadziesiąt bajtów).

Wartość MTU danych wysyłanych przez komputer nie powinna być wyższa niż MTU dowolnego urządzenia retransmitującego dane napotkanego na drodze do komputera docelowego. Jeżeli ten warunek nie jest zachowany, pakiety danych będą musiały być dzielone na mniejsze części i wysyłane we fragmentach, co zwiększa obciążenie łącza sieciowego i może spowodować znaczne spowolnienie transmisji. Utrzymaniem właściwej wartości MTU dla każdej transmisji zajmuje się na szczęście automatycznie system operacyjny.

Adresy sprzętowe

Wiele interfejsów sieciowych — na przykład urządzenia sieci Ethernet — wyposażonych jest w nadawany przez producenta adres sprzętowy, tak zwany MAC (ang. *Media Access Control*). W przypadku kart sieciowych Ethernet ma on postać sześciu dwucyfrowych liczb szesnastkowych, na przykład: *00:0D:10:65:A0:01*.



Adres sprzętowy nazywa się też często adresem fizycznym lub adresem MAC.



Do zamiany adresów IP na adresy MAC służy protokół ARP (ang. *Address Resolution Protocol*). Jest on niezbędny, gdyż karta sieciowa Ethernet może nadać dane do dowolnej innej karty, tylko podając jej adres sprzętowy. Na szczęście takie tłumaczenie odbywa się w pełni automatycznie i bez konieczności konfiguracji systemu operacyjnego.

W zamyśle adresy sprzętowe miały być unikatowe i niezmienne, obecnie jednak praktycznie każde urządzenie można przeprogramować tak, aby nadać mu dowolny adres sprzętowy (fizyczny).

Wyświetlanie listy interfejsów sieciowych

Aby wyświetlić listę interfejsów sieciowych obecnych w komputerze, wydaj — zalogowany jako administrator (`root`) — polecenie `ip a` (rysunek 3.2).

Rysunek 3.2. *Lista interfejsów sieciowych wyświetlona za pomocą polecenia ip a*

```
root@serwerek:~# ip a
1: lo: <LOOPBACK,UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
2: eth0: <BROADCAST,MULTICAST> mtu 1500 qdisc noop qlen 1000
    link/ether 00:0c:29:8b:7d:04 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop qlen 1000
    link/ether 00:0c:29:8b:7d:0e brd ff:ff:ff:ff:ff:ff
root@serwerek:~# _
```

Poszczególne wiersze uzyskane w powyższym przykładzie mają następujące znaczenie:

1. Pierwszy (`1:`) interfejs sieciowy o nazwie `lo` to interfejs typu *loop-back* (`LOOPBACK`). Jest on aktywny (`UP`), a wartość MTU wynosi dla niego 16 436 bajtów (`mtu 16436`).
2. Adres fizyczny interfejsu *loop-back* (`link/loopback`) ma wartość `00:00:00:00:00:00`. Adres rozgłaszania ma również wartość `00:00:00:00:00:00`, ponieważ w sieci typu *loop-back* publiczne rozgłaszanie pakietu danych nie ma sensu, gdyż istnieje tylko jeden komputer.



Adres rozgłaszania to adres, pod który należy nadać dane, aby odebrały je wszystkie komputery w danej sieci lokalnej. Również sieci IP dysponują takim adresem: jest to adres, w którym wszystkie bity adresu komputera ustawione są na 1. W sieci 192.168.0.0/24 przykładem może być adres 192.168.0.255.

3. Adres IP i maska interfejsu (`inet`) to 127.0.0.1 i 255.0.0.0 (`127.0.0.1/8`).
4. Drugi (`2:`) interfejs sieciowy o nazwie `eth0` może służyć do rozgłaszania informacji (`BROADCAST, MULTICAST`). Wartość MTU wynosi dla niego 1500 bajtów (`mtu 1500`).
5. Adres fizyczny interfejsu typu Ethernet (`link/ether`) ma wartość `00:0C:29:8B:7D:04`. Adres rozgłaszania ma wartość `FF:FF:FF:FF:FF:FF` (jest to wartość typowa dla wszystkich sieci Ethernet).
6. Trzeci (`3:`) interfejs sieciowy o nazwie `eth1` może służyć do rozgłaszania informacji (`BROADCAST, MULTICAST`). Wartość MTU wynosi dla niego 1500 bajtów (`mtu 1500`).
7. Adres fizyczny interfejsu typu Ethernet (`link/ether`) ma wartość `00:0C:29:8B:7D:0E`. Adres rozgłaszania — jak poprzednio — ma wartość `FF:FF:FF:FF:FF:FF`.

Aktywacja interfejsu sieciowego

Interfejs sieciowy — niezależnie od tego, czy został mu nadany adres IP — może być aktywny lub nie. W czasie uruchamiania systemu Slackware Linux automatycznie aktywowane są interfejsy, dla których nadany został adres. Jeżeli w czasie pracy chcesz wyłączyć lub włączyć jeden z interfejsów, musisz wydać polecenie:

```
ip l set interfejs up
```

aby aktywować interfejs, lub polecenie:

```
ip l set interfejs down
```

aby go wyłączyć (rysunek 3.3).

```
root@serwerek:~# ip l set eth0 up
root@serwerek:~# ip a
1: lo: <LOOPBACK,UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
2: eth0: <BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:0c:29:8b:7d:04 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop qlen 1000
    link/ether 00:0c:29:8b:7d:0e brd ff:ff:ff:ff:ff:ff
root@serwerek:~# ip l set eth1 down
root@serwerek:~# ip a
1: lo: <LOOPBACK,UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
2: eth0: <BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:0c:29:8b:7d:04 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop qlen 1000
    link/ether 00:0c:29:8b:7d:0e brd ff:ff:ff:ff:ff:ff
root@serwerek:~# _
```

Rysunek 3.3. *Po aktywowaniu interfejsu eth0 na liście pojawiło się słowo UP oznaczające jego działanie; po deaktywacji znika ono z listy*



Więcej informacji na temat usługi DNS i konfigurowania własnego serwera DNS znajdziesz w Rozdziale 7.

Konfigurację sieci — na przykład przypisanie adresów IP — możesz ustalić na stałe w jednym z plików konfiguracyjnych lub zmieniać dynamicznie w czasie pracy systemu, jeżeli zaistnieje taka potrzeba.

Ustalanie domyślnej konfiguracji sieci

Aby system automatycznie podczas uruchamiania konfigurował parametry interfejsów sieciowych typu Ethernet (czyli najpopularniejszych kart sieciowych), należy zmodyfikować plik konfiguracyjny o nazwie */etc/rc.d/rc.inet1.conf*. Edycję tego pliku rozpoczniesz, wydając polecenie `mcedit /etc/rc.d/rc.inet1.conf` (rysunek 3.4).

```
/etc/rc.d/r~net1.conf   [----] 10 L:[  1+ 0   1/ 91] *(10  /3540b)= d 100 0x64
# /etc/rc.d/rc.inet1.conf
#
# This file contains the configuration settings for network interfaces.
# If USE_DHCP[interface] is set to "yes", this overrides any other settings.
# If you don't have an interface, leave the settings null ("").

# You can configure network interfaces other than eth0,eth1... by setting
# IFNAME[interface] to the interface's name. If IFNAME[interface] is unset
# or empty, it is assumed you're configuring eth<interface>.

# Several other parameters are available, the end of this file contains a
# comprehensive set of examples.

# ============================================================================

# Config information for eth0:
IPADDR[0]=""
NETMASK[0]=""
USE_DHCP[0]=""
DHCP_HOSTNAME[0]=""

# Config information for eth1:
IPADDR[1]=""
1Help  2Save  3Mark  4Replac 5Copy  6Move  7Search 8Delete 9PullDn 10Quit
```

Rysunek 3.4. *Edycja pliku konfiguracyjnego odpowiedzialnego za początkową konfigurację IP interfejsów sieciowych, narzucaną podczas uruchamiania systemu*

Plik konfiguracyjny zawiera cztery powtarzające się sekcje czterech ustawień, odpowiadające za konfigurację interfejsów sieciowych `eth0`, `eth1`, `eth2` i `eth3`. Na przykład, aby interfejs `eth0` otrzymywał podczas uruchamiania systemu adres 192.168.0.222 i maskę 255.255.255.0 (/24), należy wprowadzić następujące zmiany:

```
IPADDR[0]="192.168.0.222"
NETMASK[0]="255.255.255.0"
USE_DHCP[0]=""
DHCP_HOSTNAME[0]=""
```

Z kolei aby interfejs eth2 pobierał adres IP automatycznie z serwera DHCP działającego w podłączonej do niego sieci (na przykład z serwera DHCP dostawcy łącza internetowego), należy wprowadzić następujące zmiany:

```
IPADDR[2]=""
NETMASK[2]=""
USE_DHCP[2]="yes"
DHCP_HOSTNAME[2]="mojserwer"
```



Nazwa interfejsu wprowadzana w polu DHCP_HOSTNAME jest zazwyczaj dowolna i czasem zbędna. Właściwe informacje na jej temat powinieneś otrzymać od administratora sieci lokalnej, do której podłączony jest dany interfejs sieciowy.

Poniżej znajduje się osobna opcja, wspólna dla całego systemu, w której powinieneś podać adres IP komputera pełniącego rolę bramy internetowej (routera). Adres ten zostanie Ci podany przez dostawcę łącza internetowego:

```
GATEWAY="192.168.0.254"
```

Zmiany zapiszesz sekwencją klawiszy *F2*, *Enter*, a edytor *mcedit* opuścisz, naciskając klawisz *F10*. Aby wprowadzone w tym pliku zmiany odniosły skutek, konieczne jest powtórne uruchomienie serwera.

Wprowadzanie adresu serwera DNS

Adres serwera DNS wprowadza się w pliku konfiguracyjnym */etc/resolv.conf*. Po wywołaniu go do edycji poleceniem `mcedit /etc/resolv.conf` usuń dotychczasową zawartość i wprowadź adres IP serwera, poprzedzając go słowem `nameserver`, na przykład:

```
nameserver 192.168.0.1
```

Możesz wprowadzić tutaj adresy kilku serwerów — dwóch, trzech czy nawet większej liczby. W takim przypadku, jeżeli pierwszy z nich nie będzie odpowiadał na zapytania, zostanie użyty drugi i tak dalej, aż do uzyskania odpowiedzi lub wyczerpania listy:

```
nameserver 157.157.3.1
nameserver 157.158.3.2
```

Jeżeli po lekturze rozdziału 7. zdecydujesz się na uruchomienie własnego serwera DNS, plik */etc/resolv.conf* powinien mieć zawartość:

```
nameserver 127.0.0.1
```

gdzie adres 127.0.0.1 odpowiada interfejsowi *loop-back*, czyli samemu serwerowi.

Zmiany wprowadzone w pliku */etc/resolv.conf* zaczynają obowiązywać automatycznie, po chwili od momentu zapisania nowej wersji pliku na dysku.

Wyświetlanie adresów IP przypisanych interfejsom

Aby wyświetlić adresy IP przypisane poszczególnym interfejsom sieciowym, powinieneś wydać polecenie `ip a` — to samo, które służy do wyświetlania listy interfejsów. Dla każdego interfejsu posiadającego adres pojawi się wiersz rozpoczynający się od słowa `inet` i prezentujący następujące informacje (rysunek 3.5):

```
root@serwerek:~# ip a
1: lo: <LOOPBACK,UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
2: eth0: <BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:0c:29:8b:7d:04 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.222/24 brd 192.168.0.255 scope global eth0
3: eth1: <BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:0c:29:8b:7d:0e brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.1/24 brd 10.0.0.255 scope global eth1
root@serwerek:~# _
```

Rysunek 3.5. *Adresy IP przypisane poszczególnym interfejsom sieciowym*

- adres IP oraz maskę (przykład: `192.168.0.222/24`);
- adres rozgłaszania (`brd` — *broadcast*) wyznaczony automatycznie przez system na podstawie adresu IP oraz maski (przykład: `brd 192.168.0.255`);
- typ adresu: publiczny (`scope global`), obowiązujący wyłącznie w sieci lokalnej (`scope link`) lub obowiązujący jedynie lokalnie na jednym komputerze (`scope host`).

Powinieneś pamiętać, że funkcjonować będą jedynie te interfejsy, które są aktywne, czyli w których pierwszym wierszu informacyjnym wyświetlanym poleceniem `ip a` znajduje się słowo `UP`.



Aby wyświetlić informacje dotyczące wyłącznie jednego interfejsu sieciowego, wydaj polecenie `ip a s` *`interfejs`*, na przykład `ip a s eth0`.

Dynamiczna konfiguracja interfejsów sieciowych

Czasami zachodzi potrzeba zmiany adresu IP przypisanego interfejsowi sieciowemu w czasie pracy systemu. System Linux umożliwia dynamiczną realizację takich zmian, w tym przypisywanie interfejsom sieciowym dowolnej liczby adresów IP (choć w domyślnej konfiguracji adres dodany jako pierwszy będzie preferowany).

Aby dodać adres IP do interfejsu sieciowego, wprowadź polecenie:

```
ip a a adres/maska dev interfejs
```

gdzie *adres/maska* to nowy adres IP wraz z maską (przykład: 10.1.1.8/24), a *interfejs* — nazwa interfejsu sieciowego (rysunek 3.6).

```
root@serwerek:~# ip a a 10.1.1.8/24 dev eth1
root@serwerek:~# ip a s eth1
3: eth1: <BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:0c:29:8b:7d:0e brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.1/24 brd 10.0.0.255 scope global eth1
    inet 10.1.1.8/24 scope global eth1
root@serwerek:~# _
```

Rysunek 3.6. *Dodawanie adresu IP*

Z kolei aby usunąć jeden z adresów, wydaj polecenie

```
ip a d adres/maska dev interfejs
```

Parametry tego polecenia są identyczne jak w przypadku dodawania adresu (rysunek 3.7).

```
root@serwerek:~# ip a d 10.0.0.1/24 dev eth1
root@serwerek:~# ip a s eth1
3: eth1: <BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:0c:29:8b:7d:0e brd ff:ff:ff:ff:ff:ff
    inet 10.1.1.8/24 scope global eth1
root@serwerek:~# _
```

Rysunek 3.7. *Usuwanie adresu IP*

Diagnostyka łączy sieciowych

W czasie konfigurowania łącz sieciowych — szczególnie tych prowadzących do Internetu — bardzo ważne jest biegłe opanowanie narzędzi służących do diagnozowania stanu łącza i sprawdzania komunikacji między serwerem a komputerami w sieci lokalnej lub w sieci dostawcy łącza internetowego. Poniżej znajdziesz opis kilku najważniejszych narzędzi tego typu.

ping

Polecenie `ping` *adres* wysyła do komputera o adresie (lub nazwie) *adres* testowy pakiet danych ICMP i oczekuje odpowiedzi. Polecenie przydatne w czasie testowania sieci lokalnych lub działania połączenia internetowego. Jest to jedno z najczęściej stosowanych poleceń diagnostycznych.

Najczęściej stosowane parametry:

- `-c` *n* — ogranicza liczbę pakietów do *n* (standardowo test połączenia przebiega bez końca, aż do momentu użycia kombinacji klawiszy *Ctrl+C*);
- `-i` *czas* — modyfikuje czas (w sekundach) upływający między kolejnymi próbami nadania pakietu testowego;
- `-n` — wyłącza możliwość zamiany adresów IP na odpowiadające im nazwy komputerów (przydatne w razie kłopotów z działaniem usługi DNS);